WSLP 2026

The Second Workshop on Sign Language Processing

Proceedings of the Workshop

October 29, 2026

©2026 Association for Computational Linguistics

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
317 Sidney Baker St. S
Suite 400 - 134
Kerrville, TX 78028
USA
Tel: +1-855-225-1962
acl@aclweb.org

ISBN 979-8-89176-488-0

Introduction

Welcome to the Second Workshop on Sign Language Processing (WSLP-2026), held in Budapest, Hungary, on 29 October 2026, co-located with the Conference on Empirical Methods in Natural Language Processing (EMNLP 2026). WSLP brings together researchers, linguists, practitioners, and members of Deaf and Hard-of-Hearing communities working at the intersection of natural language processing, computer vision, and sign linguistics. Sign languages are not visual encodings of the surrounding spoken language. They are independent natural languages in which handshape, orientation, movement, and location combine with facial expression, eye gaze, and torso shift, often at the same time. That simultaneity is linguistically ordinary and computationally inconvenient: most of the modelling toolkit inherited from spoken-language NLP assumes a linear token stream. More than 430 million people worldwide live with disabling hearing loss, and for a large part of the Deaf and Hard-of-Hearing community, sign languages are the primary means of communication. The workshop exists to give under-resourced sign languages a dedicated computational venue, and to treat sign language processing as a set of related problems: translation, recognition, spotting, corpus design, transfer, and evaluation, rather than as a single translation benchmark.

The 2026 edition of the WSLP workshop at EMNLP widens the geographic range of the main program while keeping Indian Sign Language as the shared-task language. Topics of interest this year included sign language translation for low-resource and underrepresented languages; continuous and gloss-free methods; multilingual, cross-modal, and zero-shot transfer; cultural and dialectal variation; corpus creation and annotation practice; fairness, bias, and ethics; fingerspelling and isolated recognition under sparse labels; and applications that support Deaf communities in low-resource settings.

In this edition, we accepted archival long papers (up to eight pages plus references) and short papers (up to four pages plus references), reviewed double-blind through OpenReview, for publication in the EMNLP 2026 workshop proceedings. We received 40 submissions and accepted 22 papers, an acceptance rate of 55%. Fifteen of the accepted papers are main-workshop research contributions; the rest are official shared-task system descriptions. Main workshop papers proposes new corpora and first baselines for Catalan, Uruguayan, Icelandic, Russian, and Bangla sign languages; compact gloss-free translation under realistic compute budgets; isolated recognition and fingerspelling; and a cluster of diagnostic studies on what actually transfers when a checkpoint is moved from a high-resource sign language to a low-resource one, on how annotation budgets should be spent, and on whether automatic metrics and multimodal models are measuring the signed signal at all.

In addition to the research track, WSLP-2026 hosted three shared tasks on Indian Sign Language: sentence-level translation from video or pose into English, isolated recognition over a large one-shot vocabulary, and word/sign presence prediction in continuous sentence video. Public training data were released on Hugging Face; official scoring was hosted on Codabench. Fifty-eight teams registered across the three tracks and 19 submitted official systems (translation: 23 registered, 5 submitted; presence prediction: 13 registered, 5 submitted; isolated recognition: 22 registered, 9 submitted). On the lexical tracks, the systems that report usable official scores treat the problem as retrieval or temporal alignment in a pose-embedding space. Translation submissions almost uniformly map pose sequences into a pretrained T5 decoder and obtain fluent English more readily than scores that can be trusted at the current accuracy level. We thank every participating team. The shared-task overview and the accepted system papers are included in these proceedings.

WSLP-2026 is a one-day workshop. Accepted papers are scheduled as oral presentations or posters by the programme committee. The programme further includes the shared-task session.

Acknowledgments

We would like to thank all the authors for submitting their work, the members of the programme committee for their thorough and timely reviews, and the teams that took part in the shared tasks. We are grateful to the Deaf signers, interpreters, and linguists who contributed to the corpora and evaluations reported here, and to the EMNLP 2026 workshop chairs and the ACL publication chairs for their guidance.

Workshop travel and related costs for the IIT Kanpur organizers were supported in part by the Research-I Foundation of the Department of Computer Science and Engineering, IIT Kanpur, and in part by the Anusandhan National Research Foundation (ANRF) under Grant No. ANRF/ARG/2025/009207/ENS. We gratefully acknowledge the financial support of the Science Foundation Ireland ADAPT Centre for Digital Content Technology under Grant No. 13/RC/2106. The ADAPT Centre's grant is co-funded under the European Regional Development Fund. The organizers affiliated with the University of West Bohemia in Pilsen have been exclusively supported by the grant of the University of West Bohemia, Project No. SGS-2025-011.

ISBN 979-8-89176-488-0

WSLP 2026 Organizing Committee

Sanjeet Singh, Abhinav Joshi, Pokala Dattatreya,
Keren Artiaga, Khushi Jaydeep Pujara, Tomáš Železný,
Ivan Gruber, Jakub Honzík, Sabyasachi Kamila,
Facundo Manuel Quiroga, Marek Hrúz, Mohammed Hasanuzzaman,
Ashutosh Modi

Organizing Committee

General Chairs

Sanjeet Singh, Indian Institute of Technology Kanpur (IIT Kanpur), India
Abhinav Joshi, Indian Institute of Technology Kanpur (IIT Kanpur), India
Pokala Dattatreya, Indian Institute of Technology Kanpur (IIT Kanpur), India
Keren Artiaga, ADAPT Centre, MTU, Cork, Ireland
Khushi Jaydeep Pujara, Indian Institute of Technology Kanpur (IIT Kanpur), India
Tomáš Železný, University of West Bohemia in Pilsen, Czech Republic
Ivan Gruber, University of West Bohemia in Pilsen, Czech Republic
Jakub Honzík, University of West Bohemia in Pilsen, Czech Republic
Sabyasachi Kamila, Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education, Manipal, India
Facundo Manuel Quiroga, Universidad Nacional de La Plata (UNLP), Argentina
Marek Hrúz, University of West Bohemia in Pilsen, Czech Republic
Mohammed Hasanuzzaman, ADAPT Centre, MTU, Cork, Ireland
Ashutosh Modi, Indian Institute of Technology Kanpur (IIT Kanpur), India

Program Committee

Reviewers

Shiva Ahir, Areej Ahmad, Humayun Akhtar

Rakesh Chandra Balabantaray

Nischal Reddy Chandra, Mrinal Chaturvedi, Anal Roy Chowdhury

Mrinal Devnath, Aritra Ambudh Dutta

Vinay Edula

Habib Irani

Arshia Kermani, Aman Koli, Rajeev Kumar

Shikhar Mathur, Vangelis Metsis, Kishan Kumar Mishra

Astha Swarupa Nayak

Gokul Srinath Seetha Ram, Jean Paul Nunes Reinhold, Franco Ronchetti

Karahan Sahin, Muktikanta Sahu, Murat Saraclar, Jyoti Kunal Shah, Godwin Shalom, Oscar Agustín Stanchi

Ashish Upadhyay

Table of Contents

Second Workshop on Sign Language Processing (WSLP-2026)
Sanjeet Singh, Abhinav Joshi, Pokala Dattatreya, Keren Artiaga, Khushi Jaydeep Pujara, Tomáš Železný, Ivan Gruber, Jakub Honzík, Sabyasachi Kamila, Facundo Manuel Quiroga, Marek Hrúz, Mohammed Hasanuzzaman and Ashutosh Modi 1

Findings of the Shared Tasks at the Second Workshop on Sign Language Processing (WSLP 2026)
Sanjeet Singh, Abhinav Joshi, Pokala Dattatreya, Keren Artiaga, Khushi Jaydeep Pujara, Tomáš Železný, Ivan Gruber, Jakub Honzík, Sabyasachi Kamila, Facundo Manuel Quiroga, Marek Hrúz, Mohammed Hasanuzzaman and Ashutosh Modi 11

ISLiM: A Compact End-to-End Sign Language Translation Framework
Shoaib Ahmed and Dipti Mishra 18

Whitening and Fine-Tuning Are Substitutes: One-Shot Indian Sign Language Recognition
Galiveeti Poornima and Sandeep Mathias 32

What Actually Transfers? Diagnosing Output-Projection Initialization in Low-Resource Sign Language Translation
Kai-Yang Peng, Chunki Lim and Shu-Kai Hsieh 43

LSC-CARE: An annotated dataset for medical Catalan Sign Language
Carlos Escolano, Joan Gomà Cortés, Gemma Barberà, Gerard Sant and Maite Melero 54

Metric-Learning Retrieval for Cross-Signer Isolated Sign Language Recognition
Md. Asif Ali, Md. Mosharaf Hossain Apurbo and Kazi Reyazul Hasan 70

Multiple-Instance Contrastive Alignment for Pose-Based Sign Spotting
Md. Mosharaf Hossain Apurbo, Md. Asif Ali and Kazi Reyazul Hasan 77

Isolated Sign Language Recognition for Icelandic Sign Language: Experiments in a Low-resource Setting
Finnur Ágúst Ingimundarson, Guðný Björk Þorvaldsdóttir, Mathias Müller and Sarah Ebling . 84

Using Sign Phonology to Improve Few-Shot Cross-Lingual Transfer Learning for Sign Language Recognition
Matteo Lanza 96

Recognizing Uruguayan Sign Language Glosses in Continuous Discourse: A Curated Dataset and First Baselines
Manu Bervejillo Rosá, Facundo Techera, Emilio Valle Lisboa, Ariel E. Stassi and Luis Chiruzzo 111

Coverage Is Not Enough: A Controlled Study of Annotation Selection for Low-Resource Sign Language
Gyanendra Shrestha and Liting Zhang 121

Shared-Encoder Contrastive Grounding for Pose-Based Indian Sign Language Translation
Gyanendra Shrestha and Liting Zhang 132

CARE: A Coverage-Aware Retrieval Ensemble for Isolated Sign Language Recognition
Nguyen Tu Nam and Hiroki Takahashi 141

Bukva: Russian Sign Language Alphabet
Petr Surovtsev, Alexander Nagaev, Karina Kvanchiani, Elizaveta Petrova and Alexander Kapitanov 147

Cross-Lingual Transfer of Self-Supervised Representations for Sign Language Translation: A Case Study for Turkish Sign Language
Özge Can Özel, Saliha Nur Şahin and Murat Saraclar 162

SpaMo-BdSL: Gloss-Free Bangla Sign Language Translation Using Spatial Configurations and Motion Dynamics with LLMs
Kowshik Saha, Adibah Mahmud and M Saifur Rahman 172

Zero-Shot Cross-Lingual Recognition of Sign Language Handshapes
Marcel Granero Moya, Carolina Del Corral Farrarós, Gloria Haro, Coloma Ballester and Ricardo Marques 177

HandReader: Advanced Techniques for Efficient Fingerspelling Recognition
Pavel Korotaev, Petr Surovtsev, Alexander Kapitanov, Karina Kvanchiani and Alexander Nagaev 193

Znaki: A Russian Fingerspelling Recognition Dataset
Petr Surovtsev, Pavel Korotaev, Alexander Nagaev and Alexander Kapitanov 203

TransAlign-SLT: Complementary Temporal and Semantic Supervision for Gloss-Free Sign Language Translation
Pallapu Mohan Krishna, Uppala Sanjay Kumar, M.K. Bhuyan and Debanga Raj Neog 213

Articulator-Level Cross-Lingual Transfer in Sign Language Translation
Ali Elaswad and Omar Mokhtar Gaber Saqr 224

SignaFusion@WSLP 2026: Gated Vision–Language Adjudication and Multi-Expert Fusion for Isolated Sign Language Recognition
Bolisetty Samanvitha, Ojas Srivastava, Pruthwik Mishra and Praveen Kumar Chandaliya 237

Testing Video MLLM Sensitivity to Hand-Localized Visual Detail in Chinese Sign Recognition: A Matched Visual Intervention Study
Bowen Ye, Dengfeng Yao, Siyuan Jing, Peng Jin, Hongzhe Liu and Cheng Xu 243

Second Workshop on Sign Language Processing (WSLP-2026)

Sanjeet Singh[1], Abhinav Joshi[1], Pokala Dattatreya[1], Keren Artiaga[2], Khushi Jaydeep Pujara[1], Tomáš Železný[3], Ivan Gruber[3], Jakub Honzík[3], Sabyasachi Kamila[4], Facundo Manuel Quiroga[5], Marek Hrúz[3], Mohammed Hasanuzzaman[2], Ashutosh Modi[1]

[1]IIT Kanpur, India, [2]ADAPT Centre, MTU, Cork, Ireland,
[3]University of West Bohemia in Pilsen, Czech Republic,
[4]Manipal Institute of Technology Bengaluru, India, [5]UNLP, Argentina

(sanjeet,ajoshi,pokala24,khushijp24,ashutoshm)@cse.iitk.ac.in
keren.artiaga@adaptcentre.ie, sabyasachi.kamila@gmail.com,
facundoq@gmail.com, zeleznyt@fav.zcu.cz, grubiv@fav.zcu.cz,
honzikj@fav.zcu.cz, mhruz@fav.zcu.cz, hasanuzzaman.im@gmail.com

Abstract

We organized the Second Workshop on Sign Language Processing (WSLP-2026), co-located with EMNLP 2026 in Budapest, Hungary, on 29 October 2026. The workshop continues the programme begun at WSLP 2025: to give under-resourced sign languages a dedicated computational venue, and to treat sign language processing as a set of related problems rather than a single translation benchmark. We received 40 submissions and accepted 22 papers. This overview concentrates on the 15 accepted main-workshop research papers. Those papers cover new corpora for Catalan, Uruguayan, Icelandic, Russian, and Bangla sign languages; compact gloss-free translation; isolated recognition and fingerspelling; and a cluster of diagnostic studies on what actually transfers when a model is moved from a high-resource sign language to a low-resource one. As part of the workshop, we hosted three shared tasks on Indian Sign Language, namely translation, isolated recognition, and word/sign presence prediction, which together attracted 58 registered teams, and 19 teams actively participated. The open questions have shifted from whether sign languages can be modeled at all to how annotation budgets should be allocated, which parts of a pretrained checkpoint are worth transferring, and how to evaluate systems when automatic metrics remain only weakly correlated with grammatical correctness.

1 Introduction

Sign languages are not visual encodings of the surrounding spoken language. Manual articulators such as handshape, orientation, movement, and location, combined with non-manual markers such as facial expression, eye gaze, and torso shift, often occur simultaneously (Brentari, 2019; Sinha, 2017). That simultaneity is linguistically ordinary and computationally inconvenient. Sign languages are fully developed natural languages with their own grammars, structures, and sociocultural contexts (Zeshan, 2003; Sinha, 2017). Yet sign language processing (SLP) remains substantially less developed than NLP for written and spoken languages. This is despite the fact that more than 430 million people worldwide live with disabling hearing loss (World Health Organization, 2026), and for a large part of the Deaf and Hard of Hearing (DHH) community, sign languages serve as the main mode of communication. There are several reasons for limited research, including the complexity of modeling sign language, the limited availability of corpora and benchmarks for sign languages (von Agris and Kraiss, 2010; Hilzensauer and Krammer, 2015; Ko et al., 2018; Adaloglou et al., 2021; Albanie et al., 2021; Camgöz et al., 2021; Duarte et al., 2021; Dal Bianco et al., 2022; Shi et al., 2022; Joshi et al., 2023, 2024). Although studies on ASL (Lin et al., 2023), DGS (Forster et al., 2012), CSL (Zhou et al., 2020), ISL (Joshi et al., 2023, 2024), and a few other languages have begun to appear, the volume of work remains modest. Surveys such as (Liang et al., 2023; Núñez-Marcos et al., 2023) provide a broader view of current advances and highlight the remaining gaps. There is still a lot of research attention that has not been distributed evenly across sign languages.

The First Workshop on Sign Language Processing (WSLP 2025), co-located with IJCNLP-AACL 2025 at IIT Bombay, was organized in that setting (Singh et al., 2025; Hasanuzzaman et al.,

Proceedings of the Second Workshop on Sign Language Processing, pages 1–10
October 29, 2026 ©2026 Association for Computational Linguistics

2025). It accepted 10 papers, including 2 official shared-task system descriptions, and ran 3 public Indian Sign Language tasks on translation, isolated recognition, and word-presence prediction. The second edition moves the workshop to EMNLP 2026 in Budapest and widens the geographic range of the main program while keeping Indian Sign Language as the shared-task language.

In this workshop, we received 40 papers and accepted 22 papers. Fifteen of those accepted papers are main-workshop papers; the rest are official shared-task system papers. The main focus of this paper is new resources, new languages, and a noticeable turn toward the analysis of transfer, annotation, and evaluation.

2 WSLP and Sign Language Processing

Sign languages are independent natural languages. Meaning is produced through the hands together with the face, head, and torso, and reference is often placed in signing space rather than named in sequence (Brentari, 2019; Yin et al., 2021). Classifier constructions, role shift, and non-manual grammar are ordinary parts of the system, not decorative extras. Computational work has had to learn this the hard way: pipelines borrowed from speech and text can be trained on signed video, but they do not automatically acquire the structure that makes a signed utterance grammatical.

For a long stretch the field organised itself around a short list of public corpora and a short list of tasks. RWTH-PHOENIX-Weather 2014T made continuous German Sign Language translation a standard test (Camgöz et al., 2018); CSL-Daily did similar work for Chinese Sign Language (Zhou et al., 2021); How2Sign and related collections gave American Sign Language a large continuous video resource (Duarte et al., 2021). Isolated recognition grew up in parallel, usually as large-vocabulary classification on citation-form clips. Fingerspelling, sign spotting, and text-to-sign generation appeared as satellite problems. Surveys of this period are consistent on two points: progress on the few well-instrumented languages was real, and most of the world's sign languages were left outside the comparison (Núñez-Marcos et al., 2023; Yin et al., 2021).

The last few years changed the modelling recipe more than they changed that imbalance. Gloss intermediates, once almost obligatory, are now optional; many systems map video or pose straight to spoken-language text. Skeleton keypoints have become a common input because they reduce compute and strip some signer-identifiable appearance, at the cost of dropping most facial grammar (Moryossef et al., 2020; Joshi et al., 2025). On the text side, pretrained decoders from the T5 family (Raffel et al., 2020) are widely reused. Self-supervised visual encoders trained on one high-resource sign language are beginning to be tested on others. None of these choices is settled. A fluent decoder can raise BLEU or chrF without a firm link to the signed input; a pose tracker can miss a hand for long stretches of a clip; a checkpoint that looks transferable may be carrying its gain in the output projection rather than in the visual encoder. Those are empirical questions, and they show up across languages, not only in one setting.

What remains thin is the surrounding infrastructure: comparable corpora for smaller signing communities, annotation protocols, and evaluation that a fluent signer would recognize as linguistically correct. WSLP was set up to put those pieces in the same venue.

The 2025 edition used Indian Sign Language for its shared tasks and drew on public ISL resources such as iSign and CISLR (Joshi et al., 2024, 2022). That choice supplies a reproducible measurement point; it is not the scope of the workshop. The 2026 main program, as the later sections show, is organized around many sign languages and around problem transfer, annotation budgets, fingerspelling, domain corpora, and model diagnostics that are not tied to any single language.

3 WSLP-2026: Workshop Scope and Organization

WSLP-2026 is a one-day workshop co-located with the 2026 Conference on Empirical Methods in Natural Language Processing in Budapest, Hungary, and held on 29 October 2026. The call for papers asked for work on low-resource and underrepresented sign languages; continuous and gloss-free translation; multilingual and zero-shot transfer; dialectal and cultural variation; corpus creation and annotation; fairness and ethics; and applications that support Deaf communities in low-resource settings. We accepted archival long papers (up to eight pages plus references) and short papers (up to four pages plus references), reviewed double-blind through OpenReview, for publication in the EMNLP 2026 workshop pro-

Item	Count
Papers submitted	40
Papers accepted	22
Accepted main-workshop research papers	15
SLT teams registered / participated	23 / 5
WPP teams registered / participated	13 / 5
ISLR teams registered / participated	22 / 9

Table 1: Official WSLP-2026 statistics. Participation figures count teams that submitted to a shared task, not papers.

ceedings. Non-archival extended abstracts were also permitted; they do not appear in the proceedings and are not part of the statistics. The organizing committee spans India, Ireland, the United Kingdom, the Czech Republic, and Argentina.

4 Submission and Acceptance Statistics

Table 1 records the official workshop figures. Forty papers were submitted, and 22 were accepted. Those 22 papers include both main-workshop research papers and official shared-task system descriptions. This overview discusses the 15 accepted main workshop papers cataloged for the research track.

The gap between registration and participation is large, especially in translation. We do not treat that gap as a quality judgment. Training continuous translation systems on video or pose still costs more than most workshop participants can spend. Isolated recognition drew the most completed submissions, which matches the relative maturity of retrieval and classification recipes for that task.

Compared with WSLP 2025, the 2026 main programme is larger and less centred on a single language. The first edition's eight main papers were weighted toward Indian Sign Language resources, multilingual pose translation, and glossing. The 2026 main papers still include ISL work, but Catalan, Icelandic, Uruguayan, Russian, Bangla, Taiwan, Turkish, and Chinese sign languages all appear as primary objects of study rather than transfer targets mentioned in passing.

5 Research Themes of WSLP-2026

The 15 accepted main papers fall into five overlapping groups. Table 2 gives a compact view; Section 6 discusses the papers themselves.

New corpora and domain-specific resources. Several papers release data rather than architectures. LSC-CARE is an annotated medical corpus of Catalan Sign Language. LSU-446 is a curated glossed dataset of Uruguayan Sign Language discourse. Bukva (Kvanchiani et al., 2024) and Znaki address isolated and continuous Russian fingerspelling. These resources matter because they change what the next paper on the same language can measure.

Isolated recognition in low-resource conditions. Icelandic Sign Language recognition is studied from dictionary video with public frameworks. A phonological canonicalization method improves few-shot cross-lingual isolated recognition. Zero-shot handshape recognition is attempted from American Sign Language supervision to Catalan Sign Language. The common constraint is sparse labels, often one example per type, and pose trackers that fail more often than papers usually admit.

Fingerspelling. HandReader (Korotaev et al., 2025) reports strong results on ChicagoFSWild and ChicagoFSWild+. Bukva and Znaki supply the missing Russian resources for the same problem. Fingerspelling is a small corner of signing, but it is where proper names and new vocabulary live, and it exposes temporal modelling problems that isolated lexical recognition can hide.

Gloss-free and compact translation. ISLiM is a compact keypoint-to-text model evaluated on iSign. SpaMo-BdSL adapts a spatial-and-motion translation recipe to Bangla Sign Language under a single-GPU budget. Both papers treat decoder choice and hardware limits as first-class design constraints rather than afterthoughts.

Transfer, annotation, and evaluation diagnostics. This is the theme that most clearly separates 2026 from 2025. One paper asks what part of a Uni-Sign (Li et al., 2025) checkpoint actually moves when the target is Taiwan Sign Language. Another decomposes articulator-level transfer from American and Chinese Sign Language into Indian Sign Language. A third tests whether articulatory coverage is a good rule for spending an annotation budget. A fourth asks whether video multimodal large language models use the hands at all when they answer questions about Chinese signs. These papers are not competing for a leaderboard. They are trying to stop the field from over-interpreting scores.

Theme	Papers	Languages / resources
Corpora and domain data	LSC-CARE; LSU-446; Bukva; Znaki; annotation-selection study	LSC, LSU, RSL; PHOENIX14T, Isharah
Low-resource ISLR	Icelandic ISLR; phonological few-shot transfer; zero-shot handshapes	ITM, TID and six transfer targets; ASL→LSC
Fingerspelling	HandReader; Bukva; Znaki	ASL, RSL
Gloss-free / compact SLT	ISLiM; SpaMo-BdSL; SHuBERT-to-TID	ISL/iSign, BdSL, TID
Transfer and diagnostics	Output-projection study; articulator-level transfer; annotation selection; video MLLM study	TSL; ASL/CSL→ISL; DGS/SSL; CNSL

Table 2: Research themes in the accepted WSLP-2026 main-workshop papers. A paper may belong to more than one theme.

6 Accepted Main-Workshop Papers

We discuss the accepted main papers by the themes above. Table 3 lists the fifteen papers for reference.

6.1 Corpora, Domains, and Annotation

LSC-CARE is the first annotated Catalan Sign Language dataset built for medical communication. It contains 2,784 signed segments, about 4.5 hours, produced by 11 deaf native signers on ten emergency-care topics. Segments carry time-aligned ID-glosses from a deaf native linguist and written Catalan translations from an LSC interpreter, following LSC SignBank conventions. The modelling experiments in the paper are as important as the release. A recognition-then-translation cascade beats a gloss-free video-to-text model, remains statistically invariant to signer identity.

Recognizing Uruguayan Sign Language Glosses in Continuous Discourse releases LSU-446: 17,793 sign instances over 446 glosses, drawn from interpreted discourse rather than citation-form recording. Average sign duration is 403 ms, coarticulation is heavy, and there were no prior computational models of LSU. SPOTER (Boháček and Hrúz, 2022) and MSG3D are compared as first isolated-recognition baselines; MSG3D reaches 32.7% Top-1 over 446 classes and degrades more slowly as the vocabulary grows. A separate spotting experiment on a held-out video from an unseen signer and domain, with 33.8% out-of-vocabulary items, compares a segmenter pipeline with a sliding window under a protocol that separates temporal detection from lexical recognition. The paper is a reminder that "low-resource" is not only a matter of hours. Interpreted news and citation-form laboratory video are different languages of recording, and baselines built on the latter travel poorly to the former.

Bukva and Znaki fill two gaps in Russian fingerspelling. Bukva is an isolated dactyl dataset: 3,757 videos, more than 100 samples per Russian alphabet sign including dynamic letters, collected from 155 deaf and hard-of-hearing experts through crowdsourcing. A Temporal Shift Module baseline reaches 83.6% top-1 accuracy with CPU-only real-time inference. Znaki is a continuous fingerspelling corpus: 37,252 recordings (35.4 hours) of 1,593 sequences from 16 domain categories, produced by 68 native RSL signers after a two-stage qualification procedure. Letter frequencies are matched to the Russian National Corpus. Squeezeformer and Fingerspelling-PoseNet baselines reach character-level accuracies of 0.932 and 0.886 respectively. Together the two datasets make Russian fingerspelling experimentally accessible at both the letter and the name-spelling granularity.

Coverage Is Not Enough paper asks a different resource question: given a fixed annotation budget, which clips should be labelled first? The study is retrospective on PHOENIX14T (Camgoz et al., 2018) and Isharah (Alyami et al., 2025), across gloss recognition and translation, multiple budgets, and three seeds. Geometric, active-learning, pose-derived articulatory, and privileged selectors are compared with random sampling. Articulatory coverage is generally weak and rarely beats random. A privileged gloss-diversity oracle also performs poorly. Facility-location representativeness is the strongest feasible selector in three of four largest-budget settings. The result is uncomfortable for anyone who assumed that "cover more handshapes" or "cover more gloss types" is an automatic improvement. As operationalised in this paper, those proxies are not reliable guides to annotation utility.

Paper	Focus
ISLiM: A Compact End-to-End Sign Language Translation Framework	Gloss-free keypoint SLT on iSign
What Actually Transfers? Diagnosing Output-Projection Initialization	Checkpoint analysis for Taiwan SL
LSC-CARE: An Annotated Dataset for Medical Catalan Sign Language	Medical LSC corpus and SLT baselines
Isolated Sign Language Recognition for Icelandic Sign Language (Ingimundarson et al., 2026)	First ITM ISLR baselines
Using Sign Phonology to Improve Few-Shot Cross-Lingual Transfer	Dominance canonicalization for ISLR
Recognizing Uruguayan Sign Language Glosses in Continuous Discourse	LSU-446 corpus and spotting baselines
Coverage Is Not Enough: Annotation Selection for Low-Resource SL	Annotation-budget study
Bukva: Russian Sign Language Alphabet	Isolated RSL dactyl dataset
Cross-Lingual Transfer of Self-Supervised Representations	SHuBERT adapted to TID
SpaMo-BdSL: Gloss-Free Bangla Sign Language Translation	Compact BdSL translation
Zero-Shot Cross-Lingual Recognition of Sign Language Handshapes (Granero-Moya et al., 2026)	ASL→LSC phonological handshapes
HandReader: Efficient Fingerspelling Recognition	ASL fingerspelling architectures
Znaki: A Russian Fingerspelling Recognition Dataset	Continuous RSL fingerspelling corpus
Articulator-Level Cross-Lingual Transfer in SLT	ASL/CSL→ISL transfer analysis
Testing Video MLLM Sensitivity to Hand-Localized Visual Detail	CNSL video-MLLM diagnostic

Table 3: Accepted WSLP-2026 main-workshop research papers. Official shared-task system papers are excluded.

6.2 Isolated Recognition and Phonological Transfer

Isolated Sign Language Recognition for Icelandic Sign Language is a first-systems paper in the best sense. The data come from ITM SignWiki, a bilingual dictionary resource. OpenHands and SPOTER are evaluated under several configurations. SPOTER is stronger, simpler to set up, and cheaper; the best OpenHands result takes more than twice as long. Within SPOTER, MediaPipe keypoints outperform the other pose estimators tried. OpenHands benefits from a multilingual training configuration. The paper does not claim a new architecture. It documents what current public tools actually do on a language that had no previous ISLR numbers, which is the measurement the next Icelandic paper will need.

Using Sign Phonology to Improve Few-Shot Cross-Lingual Transfer Learning starts from a concrete modelling mistake. Pose graphs treat the two hands as left and right. Which hand leads is a fact about the signer, not about the sign. The paper canonicalizes dominance by mirroring clips whose left hand carries more motion energy, then trains donors on Turkish Sign Language and transfers to six other sign languages. Canonicalized donors gain +6.01 accuracy points under full fine-tuning and +5.19 with frozen features; every canonicalized donor beats every standard one on the reported sign test. The same paper is unusually honest about tracker failure: across seven corpora, 0.12–65.1% of clips never have a hand detected, and up to 80.7% of apparently one-handed clips are tracking artifacts. That observation belongs in more recognition papers than currently contain it.

Zero-Shot Cross-Lingual Recognition of Sign Language Handshapes tries a different transfer route. Instead of moving a whole-sign embedding, it decomposes handshapes into five phonological features shared by American and Catalan Sign Language: selected fingers, flexion, spread, thumb position, and thumb contact, and decodes LSC handshapes from predicted features with a composite phonological distance. MLP, SL-GCN , and SHuBERT (Gueuwou et al., 2025) models trained on PopSign and Sem-Lex are tested on a 37-handshape LSC benchmark. After recording-format differences are harmonized, zero-shot transfer reaches 80.0% phonological-feature accuracy and 54.5% expected handshape accuracy. The practical claim is modest and useful: phonological feature inventories can serve as an annotation-light bridge when the target language has no handshape labels.

6.3 Fingerspelling Recognition

HandReader is the high-resource counterpart to Bukva and Znaki. It introduces three architectures: an RGB model with a Temporal Shift-Adaptive Module, a keypoint model with a Temporal Pose Encoder that treats keypoints as tensors for 2D and 3D convolution, and a joint RGB-plus-keypoint encoder. On ChicagoFSWild and ChicagoFSWild+, the models are the strongest among methods trained on a single training set, including a new state-of-the-art result on ChicagoFSWild+. Read next to the Russian papers, HandReader shows what fingerspelling recognition

looks like once the dataset problem is no longer the main constraint: temporal operators, modality fusion, and careful handling of variable-length sequences.

6.4 Gloss-Free and Compact Translation

ISLiM (ISL in Miniature) is a compact keypoint-based gloss-free translator. Temporal convolutions capture local motion, a Transformer encoder (Vaswani et al., 2017) models longer kinematic structure, and a pretrained T5-small decoder generates English. On iSign it reports a BLEU-4 (Papineni et al., 2002) of 5.85, described as state of the art among keypoint-based models on that corpus. A decoder-swap experiment on Phoenix-2014-T is used to argue that the sign-side and text-side modules can be exchanged independently. A mobile study reports 384 ms mean CPU latency on a consumer Android phone. We view the paper less as a claim about the best possible ISL system and more as a claim about the design space WSLP has been trying to keep open since 2025: pose input, small decoders, and deployment constraints treated as part of the research problem (Joshi et al., 2024, 2025).

SpaMo-BdSL adapts the SpaMo spatial-and-motion recipe to Bangla Sign Language. Two findings are emphasised. A 247M-parameter BanglaT5 (Bhattacharjee et al., 2023) decoder matches the translation quality of 3.7B-parameter mT0-XL on the authors' Bangla data, while translating the test set in 40 seconds rather than 2 minutes 43 seconds. Simple training changes, suggested by a large language model and used to fit the system on one consumer GPU, are reported as part of the method rather than hidden in a footnote. The paper is explicit about its polemic: the assumption that bigger multilingual decoders are automatically better excludes both underrepresented sign languages and the groups that work on them.

Cross-Lingual Transfer of Self-Supervised Representations for Sign Language Translation asks whether SHuBERT, a self-supervised encoder trained on American Sign Language, can be reused for Turkish Sign Language on the open-domain BUTID benchmark. Frozen SHuBERT matches a raw-feature setting that uses DINOv2 (Oquab et al., 2023) crops and pose; fine-tuning reaches 7.15 BLEU-4, 1.62 above a pose baseline. The conclusion is appropriately narrow. A high-resource self-supervised encoder is usable for low-resource SLT, but not as a frozen feature extractor taken off the shelf.

6.5 What Transfers, What Models See, and What We Should Annotate

What Actually Transfers? works on Taiwan Sign Language by recombining Uni-Sign checkpoints from CSL-to-Chinese and ASL-to-English. Under an untouched mT5-base, the released CSL-Daily visual state changes content-word accuracy by -0.5 points; with the CSL-Daily mT5 state it is worth 15.0 points. Follow-up interventions locate much of that sensitivity in the large untied output projection. On How2Sign, an mT5-base projection raises content-word accuracy by 25.0 points and wrong-clip sensitivity by $+14.7$ points. The authors are careful: absolute translation quality remains low, so the paper is diagnostic rather than a competitive TSL system. We think that caution is the contribution. Released sign-to-text checkpoints entangle visual pretraining with language-model state, and intact-checkpoint comparisons are easy to over-read.

Articulator-Level Cross-Lingual Transfer fine-tunes a pose-based model from American Sign Language to Indian Sign Language and then splits the effect across encoder, decoder, and articulator streams. The matched ASL encoder-decoder pair reaches 3.61 BLEU-4 on ISL; either piece alone is close to training from scratch. Content-word recall shows a sharp frequency effect: frequent content improves with data and training, rare and unseen content stays near zero even after twenty epochs and a 49% increase in training data. Comparing ASL and Chinese Sign Language sources, the apparent source-language gap shrinks when the output language is controlled. The authors treat residual articulator differences as suggestive rather than causal because the source checkpoints also differ in pretraining domain. Transfer, in this account, supplies a coupled starting point. It does not invent the long tail of the target language.

Testing Video MLLM Sensitivity to Hand-Localized Visual Detail uses 150 front-view isolated signs from NationalCSL-DP. A hand-target condition blurs tracked hand regions; a same-arm control applies the same operator to nearby arm regions. Four video multimodal large language models answer the same four-choice question in each condition. Mean accuracy is 38.17% on original video, 33.50% on hand-blurred video, and 36.67% on the arm control. The pre-specified control-versus-target contrast is $+3.17$ points with

a confidence interval that includes zero and an item-cluster p of 0.08172. The authors call the evidence directional rather than confirmatory. That is the right register. Benchmark accuracy on sign-language video questions does not, by itself, show that a model is using the hands.

7 Shared Tasks at WSLP-2026

WSLP-2026 introduced three Indian Sign Language tasks, with public training data on Hugging Face and official scoring on Codabench.

Task 1 is sentence-level translation from ISL video or pose into English. Public training comes from iSign (Joshi et al., 2024). Evaluation uses a hidden translation set. The official metrics are BLEU, ROUGE (Lin, 2004), and chrF (Popović, 2015). Task 2 is isolated sign recognition over a large ISL vocabulary under extreme one-shot conditions. Public training is drawn from CISLR and the official recognition set (Joshi et al., 2022). The target inventory used by system papers is on the order of 4,361 classes, with a 526-clip blind test. Metrics are accuracy and top-k accuracy. Task 3 is word/sign presence prediction introduced in (Joshi et al., 2024): given a sentence video and a query word, decide whether the corresponding sign occurs in the sentence. Metrics are accuracy, precision, recall, and F1.

Twenty-three teams registered for translation, and five participated. Thirteen teams registered for presence prediction, and five participated. Twenty-two teams registered for recognition, and nine participated.

We leave architectures, official scores, and comparisons among system papers to the companion shared-task overview. Two high-level observations belong here because they affect how the main program should be read. First, the shared tasks keep ISL as a public, reproducible benchmark, while the main workshop spans many languages; that division of labor is intentional. Second, several major papers use public iSign or CISLR data without adhering to the official hidden-split protocol. Those papers are research contributions, not unofficial leaderboard entries, and we have kept them in Section 6 for that reason.

8 Community and Research Trends

Looking across the main papers, we observe four trends.

The first is geographic spread: several accepted papers are on low-resource sign languages, such as Indian Sign Language, Catalan Medical Sign, Uruguayan Interpreted Discourse, Icelandic Dictionary Video, Russian Dactyl and Continuous Fingerspelling, Bangla Translation, Taiwanese and Turkish Transfer Settings, and Chinese Isolated Signs, all of which receive dedicated papers.

The second is a shift from architecture proposals toward measurement. The WSLP 2025 main papers introduced resources and models. The 2026 papers still do that, but a larger fraction of the program is about failures of common practice: annotation heuristics that do not beat random sampling, checkpoints whose gains live in the text projection, transfer that does not help rare content, and multimodal models whose sign-video answers may not depend on the hands.

The third is the continued dominance of pose, retrieval, and small text decoders. Graph and keypoint models, nearest-exemplar retrieval, SPOTER-style transformers, and T5-family decoders appear throughout. RGB and video-backbone systems are present in HandReader, and the MLLM study is the obvious case, but they are no longer the default. Privacy and computing are the usual reasons given, and they are good reasons. The cost, documented in the phonological transfer paper and implicit in several others, is the absence of facial grammar and brittle hand tracking.

The fourth is a more cautious use of large language models. In WSLP 2025, LLM paraphrasing was used as a data augmentation technique. In WSLP 2026, language models appear as decoders, as adjudication modules in recognition ensembles (in the shared tasks). They are not treated as a general solution to data scarcity. SpaMo-BdSL's comparison between BanglaT5 and mT0-XL is the clearest statement of that caution in the main program.

9 Discussion and Future Directions

Several limitations in the current literature recur and should shape the next edition.

Automatic metrics remain a weak proxy for signed grammaticality. BLEU, chrF, and top-k accuracy are the only metrics we can compute. None of them tells us whether a translation used signing space correctly, whether a recognizer distinguished two phonologically close signs that a fluent signer would never confuse, or whether a

medical cascade produced an utterance a patient would accept. LSC-CARE's oracle gap and the translation papers' own warnings about fluent but weakly grounded decoders point in the same direction. Native-signer evaluation is still rare.

Non-manual features are still mostly missing. Pose pipelines that drop the face cannot represent questions, negation, or role shift as first-class signals. Until facial and torso channels are modeled or until papers that omit them say so in the evaluation, continuous translation numbers will continue to understate the linguistic problem.

The phonological transfer paper quantifies missing hands; the presence-prediction shared task, discussed in the companion paper, independently finds that isolated query clips lose hand detections far more often than sentence clips. Any claim about one-handed signs, dominance, or spotting has to survive that artifact.

Transfer is not a substitute for target-language content. The Taiwanese projection study and the ISL articulator study agree on the unglamorous part: coupled encoder-decoder initialization helps, rare lexical items do not appear by magic, and source-language identity is easy to confound with the pretraining domain and the output-side language model state. Future shared tasks could make this easier to study by releasing frozen encoder-only features alongside full checkpoints.

Annotation policy needs the same scrutiny as architecture. If representativeness beats articulatory coverage on two standard corpora, then collection protocols that chase "hard" or "diverse" clips by pose geometry should be treated as hypotheses, not best practice.

Finally, community participation in dataset design is still thinner than the workshop's aims. Several 2026 corpora were produced with Deaf signers, interpreters, and linguists in the loop. LSC-CARE, Bukva, and Znaki are explicit about this. That is progress relative to scraped-only collections. It is not yet a co-design of tasks and evaluation. We continue to see partnership with institutions such as ISLRTC and Ashray Akruti, and with regional Deaf associations, as a requirement rather than an acknowledgment.

10 Conclusion

WSLP-2026 is the second edition of a workshop that began in 2025 as a venue for under-resourced sign languages. Held at EMNLP 2026 in Budapest on 29 October, the conference received 40 submissions and accepted 22 papers. The 15 main workshop research papers range from new corpora in Catalan, Uruguayan, and Russian sign languages to compact ISL and Bangla translation and to a set of diagnostic studies on transfer, annotation, and multimodal evaluation.

The shared tasks keep a public ISL measurement point for translation, isolated recognition, and sign presence. The main program does something different: it shows what the wider community is actually working on once a few large European, American, and East Asian corpora are no longer the only respectable test beds.

We do not think the field has solved continuous translation, and the papers do not claim to have. What we do see is a clearer research agenda. Spend annotation budgets on purpose. Measure which part of a checkpoint moved. Do not trust a decoder's fluency as evidence of visual grounding. Treat pose tracking failures as part of the linguistic dataset. And keep building corpora in languages that still have none.

Acknowledgments

We thank the reviewers for their comments and suggestions, the WSLP-2026 programme committee, the authors of the accepted papers, and the teams that took part in the shared tasks. Workshop travel and related costs for the IIT Kanpur organizers were supported in part by the Research-I Foundation of the Department of Computer Science and Engineering, IIT Kanpur, and in part by the Anusandhan National Research Foundation (ANRF) under Grant No. ANRF/ARG/2025/009207/ENS. We gratefully acknowledge the financial support of the Science Foundation Ireland ADAPT Centre for Digital Content Technology under Grant No. 13/RC/2106. The ADAPT Centre's grant is co-funded under the European Regional Development Fund. The organizers affiliated with the University of West Bohemia in Pilsen have been exclusively supported by the grant of the University of West Bohemia, Project No. SGS-2025-011.

References

Nikolas Adaloglou, Theocharis Chatzis, Ilias Papastratis, Andreas Stergioulas, Georgios Th Papadopoulos, Vassia Zacharopoulou, George J Xydopoulos, Klimnis Atzakas, Dimitris Papazachariou, and Petros Daras. 2021. A comprehensive study on deep

learning-based methods for sign language recognition. *IEEE Transactions on Multimedia*, 24:1750–1762.

Samuel Albanie, Gül Varol, Liliane Momeni, Hannah Bull, Triantafyllos Afouras, Himel Chowdhury, Neil Fox, Bencie Woll, Rob Cooper, Andrew McParland, and Andrew Zisserman. 2021. BOBSL: BBC-Oxford British Sign Language Dataset.

Sarah Alyami, Hamzah Luqman, Sadam Al-Azani, Maad Alowaifeer, Yazeed Alharbi, and Yaser Alonaizan. 2025. Isharah: A large-scale multi-scene dataset for continuous sign language recognition. *Preprint*, arXiv:2506.03615.

Abhik Bhattacharjee and 1 others. 2023. Banglanlg and banglat5: Benchmarks and resources for evaluating low-resource natural language generation in bangla. In *Findings of the Association for Computational Linguistics: EACL 2023*.

Matyáš Boháček and Marek Hrúz. 2022. Sign pose-based transformer for word-level sign language recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*.

Diane Brentari. 2019. *Sign Language Phonology*. Key Topics in Phonology. Cambridge University Press.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Necati Cihan Camgöz, Ben Saunders, Guillaume Rochette, Marco Giovanelli, Giacomo Inches, Robin Nachtrab-Ribback, and Richard Bowden. 2021. Content4all open research sign language translation datasets. In *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*, pages 1–5. IEEE.

Necati Cihan Camgöz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT.

Pedro Dal Bianco, Gastón Ríos, Franco Ronchetti, Facundo Quiroga, Oscar Stanchi, Waldo Hasperué, and Alejandro Rosete. 2022. Lsa-t: The first continuous argentinian sign language dataset for sign language translation. In *Ibero-American Conference on Artificial Intelligence*, pages 293–304. Springer.

Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro-i Nieto. 2021. How2sign: A large-scale multimodal dataset for continuous american sign language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jens Forster, Christoph Schmidt, Thomas Hoyoux, Oscar Koller, Uwe Zelle, Justus Piater, and Hermann Ney. 2012. RWTH-PHOENIX-weather: A large vocabulary sign language recognition and translation corpus. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3785–3789, Istanbul, Turkey. European Language Resources Association (ELRA).

Marcel Granero-Moya and 1 others. 2026. Zero-shot cross-lingual recognition of sign language handshapes. *arXiv preprint*.

Shester Gueuwou, Xiaodan Du, Greg Shakhnarovich, Karen Livescu, and Alexander H. Liu. 2025. Shubert: Self-supervised sign language representation learning via multi-stream cluster prediction. *Preprint*, arXiv:2411.16765.

Mohammed Hasanuzzaman, Facundo Manuel Quiroga, Ashutosh Modi, Sabyasachi Kamila, Keren Artiaga, Abhinav Joshi, and Sanjeet Singh, editors. 2025. *Proceedings of the Workshop on Sign Language Processing (WSLP)*. Association for Computational Linguistics, IIT Bombay, Mumbai, India (Co-located with IJCNLP–AACL 2025).

Marlene Hilzensauer and Klaudia Krammer. 2015. A multilingual dictionary for sign languages: "spreadthesign". In *Proceedings of the Conference*.

Finnur Ágúst Ingimundarson and 1 others. 2026. Isolated sign language recognition for icelandic sign language: Experiments in a low-resource setting. *arXiv preprint*.

Abhinav Joshi, Susmit Agrawal, and Ashutosh Modi. 2023. Isltranslate: Dataset for translating indian sign language. *arXiv preprint arXiv:2307.05440*.

Abhinav Joshi, Ashwani Bhat, Pradeep S, Priya Gole, Shashwat Gupta, Shreyansh Agarwal, and Ashutosh Modi. 2022. CISLR: Corpus for Indian Sign Language recognition. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10357–10366, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Abhinav Joshi, Romit Mohanty, Mounika Kanakanti, Andesha Mangla, Sudeep Choudhary, Monali Barbate, and Ashutosh Modi. 2024. iSign: A benchmark for Indian Sign Language processing. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10827–10844, Bangkok, Thailand. Association for Computational Linguistics.

Abhinav Joshi, Vaibhav Sharma, Sanjeet Singh, and Ashutosh Modi. 2025. PoseStitch-SLT: Linguistically inspired pose-stitching for end-to-end sign language translation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 13834–13853, Suzhou, China. Association for Computational Linguistics.

Sang-Ki Ko, Jae Gi Son, and Hyedong Jung. 2018. Sign language recognition with recurrent neural network using human keypoint detection. In *Proceedings of the 2018 Conference on Research in Adaptive and Convergent Systems*, pages 326–328, New York, NY, USA. Association for Computing Machinery.

M. Korotaev and 1 others. 2025. Handreader: Advanced techniques for efficient fingerspelling recognition. *arXiv preprint arXiv:2505.10267*.

A. Kvanchiani and 1 others. 2024. Bukva: Russian sign language alphabet. *arXiv preprint arXiv:2410.08675*.

Dongxu Li and 1 others. 2025. Uni-sign: Toward unified sign language understanding at scale. In *International Conference on Learning Representations*.

Zeyu Liang, Huailing Li, and Jianping Chai. 2023. Sign language translation: A survey of approaches and techniques. *Electronics*, 12(12).

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Kezhou Lin, Xiaohan Wang, Linchao Zhu, Ke Sun, Bang Zhang, and Yi Yang. 2023. Gloss-free end-to-end sign language translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12904–12916, Toronto, Canada. Association for Computational Linguistics.

Amit Moryossef, Ioannis Tsochantaridis, Roee Aharoni, Sarah Ebling, and Srini Narayanan. 2020. Real-time sign language detection using human pose estimation. In *European Conference on Computer Vision*. Springer.

Adrián Núñez-Marcos, Olatz Perez de Viñaspre, and Gorka Labaka. 2023. A survey on sign language machine translation. *Expert Systems with Applications*, 213:118993.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Gwilliam Szilvasy, and 1 others. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Maja Popović. 2015. chrf: Character n-gram f-score for automatic mt evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Bowen Shi, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2022. Open-domain sign language translation learned from online video. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Sanjeet Singh, Abhinav Joshi, Keren Artiaga, Mohammed Hasanuzzaman, Facundo Manuel Quiroga, Sabyasachi Kamila, and Ashutosh Modi. 2025. Overview of the first workshop on sign language processing (WSLP 2025). In *Proceedings of the Workshop on Sign Language Processing (WSLP)*, pages 1–5, IIT Bombay, Mumbai, India (Co-located with IJCNLP–AACL 2025). Association for Computational Linguistics.

Samar Sinha. 2017. *Indian Sign Language: An Analysis of Its Grammar*. Gallaudet University Press.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.

Ulrich von Agris and Karl-Friedrich Kraiss. 2010. SIGNUM database: Video corpus for signer-independent continuous sign language recognition. In *Proceedings of the LREC2010 4th Workshop on the Representation and Processing of Sign Languages: Corpora and Sign Language Technologies*, pages 243–246, Valletta, Malta. European Language Resources Association (ELRA).

World Health Organization. 2026. Deafness and hearing loss. `https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss`. Accessed: September 4, 2026.

Kayo Yin, Amit Moryossef, Julie Hochgesang, Yoav Goldberg, and Malihe Alikhani. 2021. Including signed languages in natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online. Association for Computational Linguistics.

Ulrike Zeshan. 2003. Indo-pakistani sign language grammar: a typological outline. *Sign Language Studies*.

H. Zhou, Y. Zhang, and H. Yu. 2020. Csl-daily: A dataset for chinese sign language translation. In *Proceedings of the Conference on Computational Linguistics*.

Hao Zhou, Wengang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. 2021. Improving sign language translation with monolingual data by sign back-translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1316–1325.

Findings of the Shared Tasks at the Second Workshop on Sign Language Processing (WSLP 2026)

Sanjeet Singh[1], Abhinav Joshi[1], Pokala Dattatreya[1], Keren Artiaga[2], Khushi Jaydeep Pujara[1], Tomáš Železný[3], Ivan Gruber[3], Jakub Honzík[3], Sabyasachi Kamila[4], Facundo Manuel Quiroga[5], Marek Hrúz[3], Mohammed Hasanuzzaman[2], Ashutosh Modi[1]

[1]IIT Kanpur, India, [2]ADAPT Centre, MTU, Cork, Ireland,
[3]University of West Bohemia in Pilsen, Czech Republic,
[4]Manipal Institute of Technology Bengaluru, India, [5]UNLP, Argentina
(sanjeet,ajoshi,pokala24,khushijp24,ashutoshm)@cse.iitk.ac.in
keren.artiaga@adaptcentre.ie, sabyasachi.kamila@gmail.com,
facundoq@gmail.com, zeleznyt@fav.zcu.cz, grubiv@fav.zcu.cz,
honzikj@fav.zcu.cz, mhruz@fav.zcu.cz, hasanuzzaman.im@gmail.com

Abstract

WSLP-2026 hosted three shared tasks on Indian Sign Language: sentence-level translation from video or pose into English (SLT), isolated recognition over a large one-shot vocabulary (ISLR), and word/sign presence prediction in continuous sentence video (WPP). Shared tasks were hosted on Codabench. Twenty-three teams registered for translation and five submitted; thirteen registered for WPP and five submitted; twenty-two registered for recognition and nine submitted. On the lexical tracks, the systems that report usable official scores treat the problem as retrieval or temporal alignment within a pose-embedding space, often with an external exemplar bank. Translation submissions almost uniformly map pose sequences into a pretrained T5 decoder. They obtain fluent English more readily than scores that can be trusted at the current accuracy level: official BLEU can sit at zero, while chrF remains small and sensitive to output length. The edition shows that ISL can be measured by automatic evaluation metrics, but that current scores still say more about retrieval recipes and decoder fluency than about signed grammar. A later edition should mark pose-tracker failures and place native-signer judgments alongside BLEU and chrF.

1 Introduction

Sign languages (SL) are the primary means of communication for Deaf and Hard-of-Hearing communities. SLs are not a gestural code for the surrounding spoken language. Handshape, movement, location, and orientation combine with facial expression, eye gaze, and body posture into a simultaneous signal, and much of the grammar is carried by that simultaneity rather than by linear word order (Brentari, 2019; Yin et al., 2021; Sinha, 2017; Núñez-Marcos et al., 2023). Video is expensive to store and to annotate; fluent annotators are scarce; pose estimators drop the face and sometimes the hands; and automatic metrics borrowed from text translation say little about whether a system used signing space, classifiers, or non-manuals correctly (Moryossef et al., 2020; Joshi et al., 2025; Müller et al., 2023).

A second difficulty is uneven infrastructure. Continuous translation became a standard experiment once corpora such as RWTH-PHOENIX-Weather 2014T and CSL-Daily were available (Camgöz et al., 2018; Zhou et al., 2021). Isolated recognition and fingerspelling followed a similar path on a handful of high-resource languages (Li et al., 2020; Sincan and Keles, 2020). Most other sign languages entered the literature late, with smaller public splits and few agreed test sets (Yin et al., 2021). Without a shared evaluation protocol, research or systems in the same language can be impossible to compare, and it is hard to tell whether a reported gain is an architectural result or a change in data hygiene.

We organized the WSLP shared tasks to put one such protocol on the table for Indian Sign Language (ISL). ISL is widely used in the Indian subcontinent, yet until recently it lacked publicly available sentence-level resources at a scale sufficient to make translation experiments reproducible (Joshi et al., 2024, 2022). The shortage of certified interpreters makes automated tools both practically relevant and scientifically interesting (Joshi et al., 2023, 2024). The 2026 edition of the shared task, hosted at the Second Workshop on Sign Language Processing at EMNLP 2026 in Budapest (29 October), following the inaugural edi-

Proceedings of the Second Workshop on Sign Language Processing, pages 11–17
October 29, 2026 ©2026 Association for Computational Linguistics

tion (Singh et al., 2025), hosted three problems on the public training resources.

Sentence-level translation (SLT) asks a system to produce English text from continuous ISL sign videos. Isolated sign language recognition (ISLR) asks it to name a short citation-form clip when the target vocabulary is large and almost every class has a single training example. Word/sign presence prediction (WPP) asks whether a queried sign occurs anywhere inside a continuous sentence video. Taken together, they cover the lexical, grounding, and sentence levels that a working sign-language system actually needs. Public training data come from iSign for translation and for word/sign presence prediction (Joshi et al., 2024), and from CISLR for isolated recognition (Joshi et al., 2022). This paper is the shared-task report. Here, we present the task designs, document official participation, and compare the submitted system papers.

2 Task Design

Table 1 summarises the three tracks. Tables 2, 3, and 4 report the official test-phase rankings on translation, isolated recognition, and word/sign presence prediction. Eleven system papers were submitted in the shared-task catalog: six were accepted into the program, and five were not included. In this paper, we will discuss all the papers and technical reports submitted.

2.1 Formal task definitions

We denote a signed input as a variable-length sequence $x = (x_1, \ldots, x_T)$, where each x_t is either an RGB frame or a pose vector (body, hand, and, when available, face landmarks). Both modalities were released; participants were free to use either or both. Let $\mathcal{X}$ be the space of such sequences, $\mathcal{Y}$ the space of English strings, and $\mathcal{V}$ a finite inventory of word or gloss labels.

Sign language translation. Given a continuous sentence-level sequence $x \in \mathcal{X}$, the system must generate a token sequence $\hat{y} = (\hat{y}_1, \ldots, \hat{y}_M) \in \mathcal{Y}$. The intended mapping is

$$f_{\text{SLT}} : \mathcal{X} \to \mathcal{Y}, \qquad \hat{y} = f_{\text{SLT}}(x). \tag{1}$$

Training pairs (x, y) come from iSign (Joshi et al., 2024). At test time x is drawn from a hidden collection of source videos that is disjoint from the public training set. Official scores are corpus-level BLEU, ROUGE, and chrF between $\hat{y}$ and English reference y (Papineni et al., 2002; Lin, 2004; Popović, 2015).

Isolated sign language recognition. Given a short citation-form clip $x \in \mathcal{X}$ that contains one lexical sign, the system must return a label $\hat{v} \in \mathcal{V}$,

$$f_{\text{ISLR}} : \mathcal{X} \to \mathcal{V}, \qquad \hat{v} = f_{\text{ISLR}}(x). \tag{2}$$

The natural decision rule is nearest-exemplar retrieval in an embedding space (Joshi et al., 2022; Selvaraj et al., 2022). Evaluation is top-k accuracy on a 526-clip blind test.

Word/sign presence prediction. Given a continuous sentence sequence $x \in \mathcal{X}$ and a query word $w \in \mathcal{W}$ drawn from the official presence set built on iSign sentence video (Joshi et al., 2024), the system must decide whether the corresponding sign occurs anywhere in x:

$$f_{\text{WPP}} : \mathcal{X} \times \mathcal{W} \to \{0, 1\}, \qquad \hat{z} = f_{\text{WPP}}(x, w). \tag{3}$$

The official metrics are accuracy, precision, recall, and F1 on $\hat{z}$ against a binary label z.

The three maps are related but not interchangeable. ISLR names an isolated clip; WPP asks whether that lexical item is realized inside a longer utterance; SLT requires a fluent spoken-language string for the whole utterance. A system that solves one does not automatically solve the others.

3 Translation

Two single-task translation papers were cataloged, plus one multi-task paper that reports a translation number.

Shared-Encoder Contrastive Grounding projects pose sequences into a pretrained T5 encoder–decoder and, during training, encodes the target sentence with the same T5 encoder. An auxiliary contrastive loss aligns pose and text representations beside the usual translation objective. Clip-level normalization is used to reduce scale artifacts. On a video-held-out iSign development split (Joshi et al., 2024), contrastive grounding raises chrF (Popović, 2015) from 0.102 to 0.121, and normalization takes it to 0.128. The official submission reports chrF 0.16. The same paper shows that official chrF moves with output length and with English content that does not depend on the input.

Investigating Temporal Motion Features feeds pose through a lightweight encoder into T5-small, T5-base, and T5-large (Raffel et al., 2022), and adds a T5-small variant that explicitly models frame-to-frame pose differences. T5-small is

Task	Input → output	Public data	Metrics
SLT	ISL video/pose → English sentence	iSign (118k pairs) + hidden set	BLEU, ROUGE, chrF
ISLR	isolated clip → word label	CISLR + official recognition set	Accuracy, top-k
WPP	(sentence video, word) → present/absent	official presence set	Accuracy, P/R/F1

Table 1: WSLP-2026 shared tasks. Official scoring was hosted on Codabench.

#	Team	chrF
1	slp_sign	0.17
2	gyanendra312	0.16
3	anhnamxtanh	0.15
4	thekraken	0.14
5	manav_dhamecha	0.13

Table 2: SLT test phase, ranked by chrF.

#	Team	Top-1
1	slp_sign	0.728
2	Wazz	0.700
3	varaprabha	0.683
4	ijsboy	0.671
5	Noob_Squad	0.671
6	WSLP-Task2	0.662
7	anhnamxtanh	0.656
8	gauandjosh	0.574
9	iiitdmk-nlp	0.196

Table 3: ISLR test phase, ranked by Top-1 accuracy.

strongest on BLEU and ROUGE among spatial-only models; T5-large is strongest on chrF; motion features give the largest BLEU gain in the authors' development study. On the official test set, the submitted system reports chrF 0.13, ROUGE-L 0.07, and BLEU-4 0.00, and states a fifth-place leaderboard finish. A BLEU-4 of zero with non-zero chrF is evidence that n-gram overlap with the references is vanishingly rare.

Different Scales, Different Signals (multi-task) uses a language-conditioned generator for translation and reports chrF 0.17. The technical report proposes separate pathways for recognition, spotting, and translation rather than a single pooled representation.

Observations: Every translation system we received is pose-to-T5. No submitted paper describes a competitive RGB video translator on the official hidden split. Motion features and contrastive grounding are the two concrete attempts to stop the decoder from ignoring the sign stream. Both help on development data. Neither produces translation quality that the authors themselves treat as reliable. We therefore read SLT 2026 as a diagnostic year: the task is well specified, participation is thin relative to registration, and the main result is that current automatic scores sit in a range where ranking systems are fragile.

#	Team	F1
1	slp_sign	0.980
2	apurbo57	0.720
3	anhnamxtanh	0.669
4	Yaswitha	0.605
5	thekraken	0.546

Table 4: WPP test phase, ranked by macro-F1.

4 Isolated Recognition

Whitening and Fine-Tuning Are Substitutes uses MediaPipe keypoints (MediaPipe, 2023), a graph-attention backbone pretrained on a different ISL corpus, PCA whitening, and nearest-exemplar cosine retrieval against an enriched gallery. The official scores reported in the paper are 0.6825 top-1 (3rd) and 0.7357 top-10 (2nd). The analytical claim is sharper than the ranking. PCA whitening of a frozen cross-corpus encoder is worth $+18.85$ points on a leave-one-out gallery evaluation; end-to-end fine-tuning is worth about the same; the two do not add. Once the encoder is fine-tuned, whitening contributes $+0.38$. Fifteen other interventions are reported as null or harmful, with McNemar tests (McNemar, 1947) on a 2,030-query held-out split. The authors argue that the 526-clip official test set is too small for the comparisons participants want to make on it.

Metric-Learning Retrieval for Cross-Signer ISLR pretrains a metric encoder on multi-signer American Sign Language, then fine-tunes on every responsible cross-signer ISL recording the authors can add. Multiple performances inside one recording are split into separate samples. A missing-keypoint interpolation method, originally developed for classification, is adapted here for retrieval. Official test scores in the paper are 67.11% top-1, 69.20% top-5, and 70.34% top-10.

CARE (Coverage-Aware Retrieval Ensemble) keeps a complete local visual dictionary and adds a CISLR (Joshi et al., 2022) reference bank for labels that exist in both vocabularies. A frozen Video-Swin encoder (Liu et al., 2022) ranks local labels; frozen I3D (Carreira and Zisserman, 2017) and SignRep encoders supply external evidence after orthogonal Procrustes alignment on shared-class centroids. External evidence is masked to supported labels. No target-specific backbone

fine-tuning and no large-vocabulary classification head are used. CARE reports 74.31/76.15/76.15 on the validation split and 65.59/67.11/68.25 on the official test split. CARE's authors read the 8.72-point top-1 drop as a caution against model selection on only 109 validation queries.

SignaFusion establishes a primary vision prediction, then calls auxiliary appearance, pose, and vision-language experts only when the primary model is uncertain. A disagreement-gated Qwen2.5-VL (Bai et al., 2025) module is one of those auxiliaries. Fusion variants are combined with reciprocal rank fusion. Official blind-test scores in the paper are 66.16% top-1, 72.43% top-5, and 73.19% top-10.

ContextVQ-Mamba trains a pose-only discrete tokenizer (articulator-aware graph encoder, bidirectional Mamba blocks (Gu and Dao, 2023), EMA codebook) and then recognises with frozen representations only. Because almost every official class is a singleton, the authors build a 25-class multiplicity-aware probe from the few labels that have genuine repeats. Token-plus-canonical-pose fusion reaches 28.57% top-1 on that probe. The paper is an analysis of functions as an analysis of representation complementarity rather than a full-vocabulary official submission.

Spatiotemporal CNN Ensembles compare MediaPipe kNN with pretrained R3D-18, R(2+1)D-18, and MC3-18 RGB networks. Landmark kNN obtains 4.56% top-1; R3D-18 reaches 16.54%; ensembles reach 17.68–18.63%. The strongest zero-shot submission compares MediaPipe kNN (MediaPipe, 2023) with pretrained R3D-18, R(2+1)D-18, and MC3-18 RGB networks (Tran et al., 2018).

The multi-task paper reports 0.7281 top-1 for isolated recognition, with protocol notes intended to separate visual transfer from gallery overlap.

Observations: The working ISLR systems in this edition are retrievers, not 4,361-way classifiers. External data, another ISL corpus, ASL pretraining, and CISLR as a reference bank are used by every accepted paper. Appearance and vision-language experts help when they are gated; they do not replace pose or local galleries. We also observe a spread in reported official top-1 roughly from the high teens (RGB ensembles without target adaptation) to the high 60s (adapted retrieval). That spread is large enough to be informative even if the 526-clip test set is too small for fine rankings.

5 Word/Sign Presence Prediction

Multiple-Instance Contrastive Alignment is a 2.86M-parameter pose Siamese encoder. Isolated query and sentence clip are embedded in one space, a frame-to-frame cosine map is formed, and multiple-instance pooling reduces the map over temporal windows. Training combines binary cross-entropy with MIL-NCE (Miech et al., 2020). The paper reports 72.69 accuracy and 71.95 macro-F1 on the blind test set, an improvement over the pose baseline's 61.66 accuracy and 58.42 macro-F1 on the same data. Ablations attribute the gain to contrastive alignment: without MIL-NCE the two clips do not share a metric space. The same paper documents a label-balance property of the released splits that, on the authors' analysis, fixes 76.3% of blind-test labels without a model, and asks future editions to change the split construction.

Active-Frame Filtering and Cross-Architecture Ensembling starts from a diagnostic: isolated query videos miss a hand detection in 49.3% of cases against 2.8% for continuous sentence videos. The system filters to active frames and ensembles a recurrent encoder with a cross-attention Transformer. Official test macro-F1 is 60.51%. Two of four cumulative design decisions are statistically robust on the authors' validation split; two are not, and the paper reports both.

The multi-task paper reports 0.9801 macro-F1 for presence prediction and flags evaluation identities that can inflate that number if protocols are not separated.

Observations: WPP is the task where preprocessing and split design matter as much as the encoder. Missing hands on isolated queries, not sentence video, dominate one diagnostic. Label balance in the released splits dominates another. A lightweight pose aligner with an explicit similarity map outperforms a generic sequence classifier in the accepted paper, which is consistent with the retrieval view that already governs ISLR.

6 Cross-Task Analysis

Two patterns cut across tracks. **1) Pose is the default input; appearance is an expert, not a base model.** Translation and WPP systems in the catalog are pose-first. Recognition systems that score well on the official vocabulary use pose or mixed-gallery settings and call RGB or vision-language models only for uncertain queries or for lexically

matched external references. The RGB ensemble that stays within generic video pretraining remains far below that of adapted pose retrieval. **2) Alignment objectives beat pooled classification when supervision is thin.** Contrastive grounding in SLT, cosine retrieval in ISLR, and MIL-NCE in WPP are the same idea at three timescales: put the signed input and the linguistic target in one space, then read a similarity rather than a 4,361-way or open-vocabulary softmax. Where teams kept a standard classification or generation head and hoped scale would help, official numbers are weaker or, in translation, hard to trust.

Participation itself is uneven. Recognition produced the most completed systems; translation produced the fewest relative to registrations. We believe that reflects compute and data movement more than community interest.

7 Lessons and Future Directions

The three tasks were meant to serve as a point of measurement, not a complete account of Indian Sign Language. What the submissions actually returned, once the leaderboard rank is set aside, is a short list of places where the measurement itself got in the way, and a longer list of linguistic problems that no official metric touched.

Start with translation. Several systems could produce English that reads as English. That is not the same as reading the signed sentence. A pretrained decoder will happily finish a fluent clause from a few content words, or from habits it learned on text, and BLEU-4 (Papineni et al., 2002) near zero with a non-zero chrF (Popović, 2015) is what that looks like on a hidden test set. If we run SLT again, the official suite has to stop treating decoder fluency as success. Length-controlled chrF (Popović, 2015) and content-word recall would already make small ranking gaps less theatrical. A small public subset scored by native signers would do something the automatic numbers cannot: say whether the output is an utterance a signer would accept, not only a string that shares characters with a reference. Until that exists, we should treat SLT scores as diagnostic, not as a claim that sentence-level ISL translation has been solved.

In recognition, a 4,361-class problem with almost one clip per class cannot be tuned on 109 validation queries and then ranked, to two decimal places, on 526 test clips. A larger public development gallery, or a frozen-encoder track with the backbone held fixed, would let the argument shift toward retrieval design whitening, galleries, and external references rather than overfitting the smallest split.

Presence prediction exposed the release as much as the models. If label balance can settle most of the blind-test items before a frame is read, the split is doing part of the classification. Isolated query clips and sentence clips also do not arrive in the same condition under current pose estimators (MediaPipe, 2023; Cao et al., 2019): hands are far more often missing in the queries. Shipping detection-quality flags with the data is not a courtesy. Without them, a spotting paper cannot tell a one-handed sign from a tracker failure, and a linguistic claim about dominance or classifiers is not available to make. Future WPP splits should be balanced so that the video is necessary, and they should mark when the pose is simply not there.

In the next edition, it should look less like three isolated contests and more like a small map of that grammar. Keep pose; do not pretend it is the whole signal. Give rare lexical items and productive constructions a place in the evaluation, not only the frequent content words that a decoder can guess.

8 Conclusion

The 2026 WSLP hosted three shared tasks on Indian Sign Language: Translation, Recognition, and word/gloss presence prediction. Fifty-eight registrations produced nineteen official submissions. Translation is still expensive to finish; recognition is the track most teams can complete. The system papers and technical reports, in the program, tell a consistent story about the method. ISLR and WPP work when the signed input and the linguistic target are placed in one space and read by similarity: nearest-exemplar retrieval, coverage-aware galleries, frame-to-frame maps, multiple-instance contrastive loss. SLT work in this edition feeds the pose modality into T5 (Raffel et al., 2022), using contrastive grounding or explicit motion as the main attempt to prevent the decoder from ignoring the hands. The official translation numbers remain in a range where the ranking systems are fragile. Pose is still the right default input for privacy and cost. It is not a full linguistic representation, and the official data should say so: mark tracker failures, keep non-manual channels in view, and put a small native-signer study next to BLEU and chrF.

Acknowledgments

We thank the reviewers for their comments and suggestions, the WSLP-2026 programme committee, the authors of the accepted papers, and the teams that took part in the shared tasks. Workshop travel and related costs for the IIT Kanpur organizers were supported in part by the Research-I Foundation of the Department of Computer Science and Engineering, IIT Kanpur, and in part by the Anusandhan National Research Foundation (ANRF) under Grant No. ANRF/ARG/2025/009207/ENS. We gratefully acknowledge the financial support of the Science Foundation Ireland ADAPT Centre for Digital Content Technology under Grant No. 13/RC/2106. The ADAPT Centre's grant is co-funded under the European Regional Development Fund. The organizers affiliated with the University of West Bohemia in Pilsen have been exclusively supported by the grant of the University of West Bohemia, Project No. SGS-2025-011.

References

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Diane Brentari. 2019. *Sign Language Phonology*. Key Topics in Phonology. Cambridge University Press.

Necati Cihan Camgöz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT.

Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. 2019. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Albert Gu and Tri Dao. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*.

Abhinav Joshi, Susmit Agrawal, and Ashutosh Modi. 2023. ISLTranslate: Dataset for translating Indian Sign Language. In *Findings of the Association for Computational Linguistics: ACL 2023*, Toronto, Canada. Association for Computational Linguistics.

Abhinav Joshi, Ashwani Bhat, Pradeep S, Priya Gole, Shashwat Gupta, Shreyansh Agarwal, and Ashutosh Modi. 2022. CISLR: Corpus for Indian Sign Language recognition. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10357–10366, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Abhinav Joshi, Romit Mohanty, Mounika Kanakanti, Andesha Mangla, Sudeep Choudhary, Monali Barbate, and Ashutosh Modi. 2024. iSign: A benchmark for Indian Sign Language processing. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10827–10844, Bangkok, Thailand. Association for Computational Linguistics.

Abhinav Joshi, Vaibhav Sharma, Sanjeet Singh, and Ashutosh Modi. 2025. PoseStitch-SLT: Linguistically inspired pose-stitching for end-to-end sign language translation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 13834–13853, Suzhou, China. Association for Computational Linguistics.

Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. 2020. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, Barcelona, Spain. Association for Computational Linguistics.

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. 2022. Video swin transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3202–3211.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

MediaPipe. 2023. MediaPipe Holistic.

Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 2020. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9879–9889.

Amit Moryossef, Ioannis Tsochantaridis, Roee Aharoni, Sarah Ebling, and Srini Narayanan. 2020. Real-time sign language detection using human pose estimation. In *European Conference on Computer Vision*. Springer.

Mathias Müller, Zifan Jiang, Amit Moryossef, Annette Rios, and Sarah Ebling. 2023. Considerations for meaningful sign language machine translation based

on glosses. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Toronto, Canada. Association for Computational Linguistics.

Adrián Núñez-Marcos, Olatz Perez de Viñaspre, and Gorka Labaka. 2023. A survey on sign language machine translation. *Expert Systems with Applications*, 213:118993.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2022. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21.

Prem Selvaraj, Gokul Nc, Pratyush Kumar, and Mitesh Khapra. 2022. OpenHands: Making sign language recognition accessible with pose-based pretrained models across languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Dublin, Ireland. Association for Computational Linguistics.

Ozge Mercanoglu Sincan and Hacer Yalim Keles. 2020. Autsl: A large scale multi-modal turkish sign language dataset and baseline methods. *IEEE Access*, 8.

Sanjeet Singh, Abhinav Joshi, Keren Artiaga, Mohammed Hasanuzzaman, Facundo Manuel Quiroga, Sabyasachi Kamila, and Ashutosh Modi. 2025. Overview of the first workshop on sign language processing (WSLP 2025). In *Proceedings of the Workshop on Sign Language Processing (WSLP)*, pages 1–5, IIT Bombay, Mumbai, India (Co-located with IJCNLP–AACL 2025). Association for Computational Linguistics.

Samar Sinha. 2017. *Indian Sign Language: An Analysis of Its Grammar*. Gallaudet University Press.

Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. 2018. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6450–6459.

Kayo Yin, Amit Moryossef, Julie Hochgesang, Yoav Goldberg, and Malihe Alikhani. 2021. Including signed languages in natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online. Association for Computational Linguistics.

Hao Zhou, Wengang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. 2021. Improving sign language translation with monolingual data by sign back-translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1316–1325.

ISLiM: A Compact End-to-End Sign Language Translation Framework

Shoaib Ahmed and Dipti Mishra
Department of Computer Science & Engineering
MU-VT IARCT[2]
Mahindra University, Hyderabad, India
{se25pcse006, dipti.mishra}@mahindrauniversity.edu.in

Abstract

Sign Language Translation (SLT) is a key accessibility technology for Deaf communities. Although recent years have seen rapid progress in SLT, Indian Sign Language (ISL) and other sign languages of under-resourced regions remain under-researched and overshadowed by sign languages with a longer research history and stronger infrastructure. In this direction, we present ISLiM (ISL in Miniature), a compact end-to-end keypoint-based gloss-free SLT architecture that maps body-pose landmark sequences directly to spoken-language text. At its core, a stack of temporal convolutional layers captures low-level pose motion patterns, a Transformer-based encoder models higher-order kinematic interactions and sign-level temporal ordering, and a pretrained T5-small decoder with a prior grasp of the target language performs the semantic heavy-lifting of generating fluent text. We further explore complementary loss functions that sustain training on small datasets, and alternative qualitative evaluation metrics tailored to the particular challenges of SLT. ISLiM achieves a BLEU-4 score of 5.85 on iSign, establishing state-of-the-art results among reported keypoint-based models on the only large-scale ISL dataset. We also show promising transferability to German Sign Language on Phoenix-2014-T through modular decoder swap, pointing toward cross-lingual extensibility where both sign and spoken languages can be swapped independently. A preliminary mobile deployment study confirms that the full keypoint-based model runs on a consumer Android smartphone with ∼384 ms mean inference latency on CPU, demonstrating a practical path toward offline, accessible assistive tools for the Deaf community in India and the world. Code is available at https://github.com/ahmedshoaib/ISLiM.

1 Introduction

Sign languages are the primary mode of communication for over 70 million Deaf people worldwide

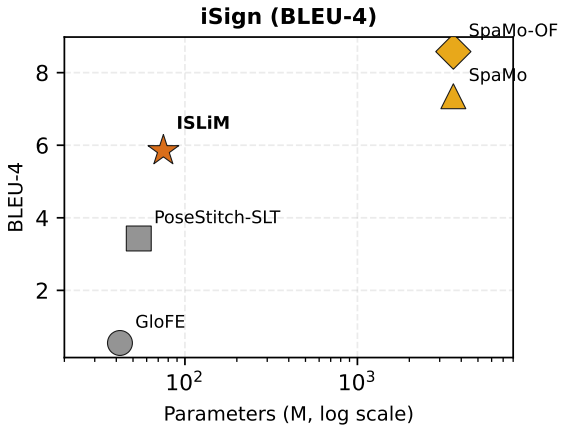


Figure 1: Parameter efficiency on iSign. ISLiM (75M, red star) improves reported keypoint SOTA by 71% (5.85 vs. 3.43). Keypoint-based methods shown as circles/squares; video-based SpaMo/SpaMo-OF as triangles/diamonds (SpaMo/SpaMo-OF only tested on 10k subset)

(WHO, 2026). India alone is home to roughly 7 million Deaf (ISLRTC, 2024a) and hard-of-hearing individuals, around 10% of the global population, yet only about 300 registered interpreters serve this community nationwide (ISLRTC, 2024b). This chronic shortage leaves most Deaf Indians without equitable access to education, employment, healthcare, and broader civic participation.

Recent advances in deep learning try to fill this shortage through Sign Language Translation (SLT), the task of mapping sign videos to spoken-language text. Large-scale SLT corpora such as iSign (Joshi et al., 2024), PHOENIX-2014-T (Camgoz et al., 2018a), How2Sign (Duarte et al., 2021), OpenASL (Shi et al., 2022), BOBSL (Albanie et al., 2021), and Multi-VSL (Dinh et al., 2025) have powered strong SLT systems. ISL, however, has just one large-scale dataset, iSign, reflecting a broader asymmetry: ISL and other sign languages of under-resourced regions remain under-researched and overshadowed by sign languages with longer re-

Proceedings of the Second Workshop on Sign Language Processing, pages 18–31
October 29, 2026 ©2026 Association for Computational Linguistics

search history and far larger datasets (Ni et al., 2026).

Existing SLT methods have evolved along two main branches. Earlier work relies on gloss-level supervision (Camgöz et al., 2020; Yin and Read, 2020; Zhou et al., 2021; Chen et al., 2022; Alvarez et al., 2022), which provides a structured intermediate target but is costly to annotate, a particular burden for low-resource settings such as ISL where few glossed resources or qualified interpreters exist. A gloss-free approach learns a direct mapping from signed utterances to text (Zhou et al., 2023; De Coster and Dambre, 2022; Li et al., 2025) by leveraging pretrained vision-language encoders and language models. Keypoint-based approaches further improve practicality: pose landmarks are privacy-preserving, lightweight to extract, and offer an explicit geometric encoding of body-part locations that are central to sign language conveyance (Lin et al., 2023). More recently, large language models have entered the SLT space (Gong et al., 2024; Wong et al., 2024; Chowdhury and Sanyal, 2025) and promise strong translation quality, but their training and inference are compute-heavy and typically assume continuous network access, assumptions that do not hold for many Indian users with limited connectivity and low-end devices (Kumar et al., 2022).

We address these constraints with ISLiM (ISL in Miniature), a compact end-to-end keypoint-based gloss-free SLT architecture. ISLiM pairs a small visual encoder with a pretrained T5-small (Raffel et al., 2020) decoder that already understands English structure and supplies much of the semantic heavy-lifting the encoder alone cannot provide. Each frame's landmark vector passes through stacked 1D temporal convolutions (Lea et al., 2017) that capture local pose-motion patterns, then a multi-layer self-attention Transformer (Vaswani et al., 2017) integrates higher-order kinematic interactions and resolves sign-level temporal ordering before the frame-level representations flow into T5 cross-attention, which produces autoregressive English output. This division of labor lets the lightweight encoder (14.5M of 75M total parameters) focus on visual feature extraction while the pretrained decoder handles fluency and word ordering, keeping the model compact enough for edge deployment. Figure 1 visualizes the parameter–quality trade-off for iSign: it attains BLEU-4 5.85 on our maintained split, against a published 3.43 for the prior keypoint-based state of the art at a comparable parameter scale. Because the baselines were evaluated under their own undisclosed splits, the comparison is indicative.

ISL presents a unique set of challenges as compared to other Sign languages, each of which informs a specific design choice in our architecture. Its vocabulary exhibits significant regional variation, emerging from a patchwork of local educational traditions (Jayachandran*, 2026). Consequently, the encoder must remain robust to multiple valid signs of the same gloss. We therefore employ a Transformer encoder, whose self-attention mechanism operates over temporally ordered frame-level representations rather than individual keypoint tokens; this allows the encoder to model long-range temporal dependencies directly, without the neighbourhood-constrained message passing of graph-based encoders (Yan et al., 2018), complementing the local motion patterns captured by the temporal convolutions that precede it. ISL distinguishes many signs using subtle contact locations on the face and torso (Sinha, 2018). To preserve these distinctions, each frame is represented using an expanded set of 94 landmarks, including 46 facial landmarks, together with estimated depth (z) and landmark visibility. This richer representation captures facial expressions, head pose, and mouth movements, which encode grammatical information such as negation, interrogatives, and topicalization that cannot be inferred from hand trajectories alone. Crucially, ISL belongs to the SOV family while English follows SVO order (Napoli and Sutton-Spence, 2014; Sugandhi et al., 2020): verbs, question markers, and negation appear late in a signed utterance but must reorder to earlier English positions (Ghosh and Mamidi, 2022). A faithful gloss-free translation therefore cannot be produced incrementally as the video plays; the model must observe the entire sign sequence before committing to any word of the English output. Our Transformer-based encoder layers learn this non-monotonic alignment (e.g., "I school go" becomes "I go to school"; "Your name what?" becomes "What is your name?") before any English token is emitted, motivating a global-attention architecture over a recurrent or frame-by-frame approach.

Our main contributions are:

- We introduce ISLiM, a compact 75M-parameter keypoint-based gloss-free end-to-end SLT architecture, achieving a test BLEU-4 score of **5.85** and establishing a new state

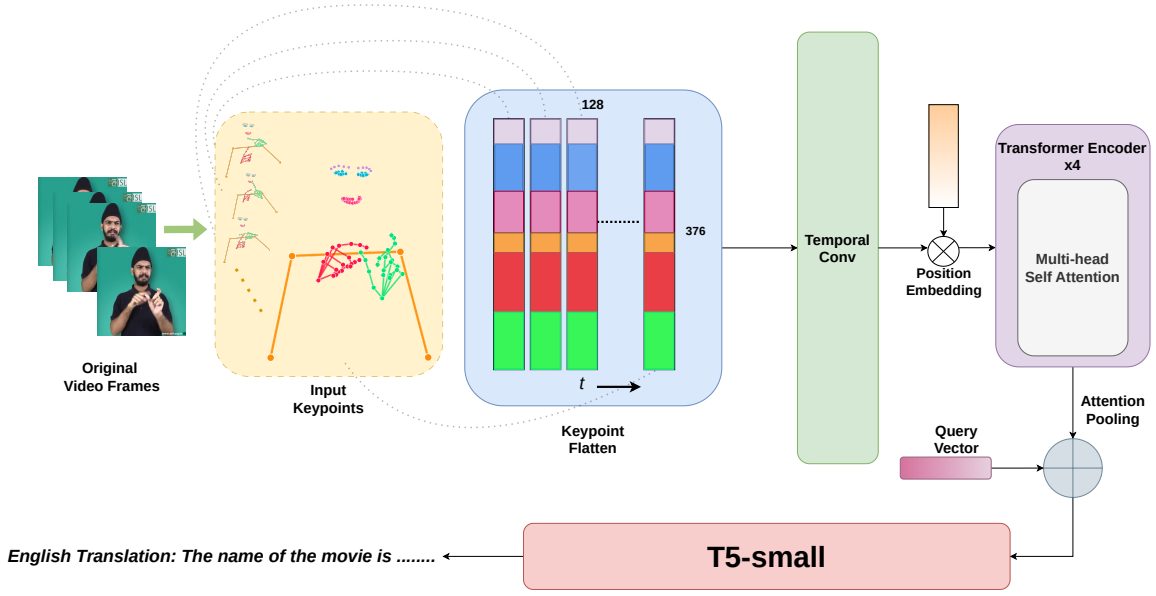


Figure 2: **ISLiM architecture overview.** The sequence of 128 frame-level representations is passed through the TCN and Transformer Encoder to the T5-small model, whose decoder generates the English translation autoregressively

of the art on iSign among reported keypoint-based models.

- We show that a cross-lingual transfer setup achieves promising results on Phoenix-2014-T (German Sign Language) at **BLEU-4 14.99**, pointing toward a modular architecture supporting multi-lingual support.
- We propose a multi-loss objective training strategy that combines cross-entropy with CLIP contrastive, semantic similarity, and entropy-regularization losses, preventing the collapse observed with single-loss baselines on smaller iSign subsets.
- We conduct a preliminary on-device deployment study on a consumer Android smartphone, reporting **384 ms** mean CPU inference latency, confirming offline feasibility without cloud connectivity for keypoint-based models.

2 Related Work

2.1 Gloss-Based Sign Language Translation

Traditional SLT systems have relied on an intermediate gloss representation, either as a necessity (Camgöz et al., 2020; Yin and Read, 2020; Zhou et al., 2021) or a desirable (Chen et al., 2022; Alvarez et al., 2022). However, manual gloss annotation is costly, and gloss vocabularies often vary across sign languages, datasets, and annotation conventions, limiting their reuse (Colonna et al., 2026). These challenges have motivated end-to-end gloss-free SLT approaches, especially for low-resource sign languages such as ISL, where obtaining large-scale gloss annotations is impractical due to the limited availability of annotated datasets and expert sign language annotators.

2.2 Gloss-Free Sign Language Translation

Gloss-free SLT, also called end-to-end SLT, learns a direct mapping from sign video to spoken language text without requiring intermediate gloss labels (Zhou et al., 2023; De Coster and Dambre, 2022; Li et al., 2025). While the annotations in this mode are weaker than their Gloss-based counterparts (Ye et al., 2024), this paradigm has gained momentum as large-scale datasets and video-language models have become available.

Video-Based Gloss-Free Models Several recent gloss-free systems (Ye et al., 2024; Mao et al., 2026) employ large video encoders pretrained on massive vision-language corpora, combined with powerful frozen language model decoders. These approaches typically use architectures such as CLIP (Contrastive Language-Image Pre-Training) (Radford et al., 2021; Jiang et al., 2024) and Video Masked Auto Encoder (VideoMAE) (Tong et al., 2022; Chowdhury and Sanyal, 2025) to encode sign video frames, then condition a pretrained text

decoder on the visual features through adapter modules or cross-attention (Asasi et al., 2025; Hwang et al., 2025).

Keypoint-Based Gloss-Free Models An alternative approach uses skeletal keypoints rather than raw video as the input modality (Lin et al., 2023). Keypoint-based approaches extract body pose landmarks from each video frame using standard pose estimation tools, then process the resulting coordinate sequences with lightweight neural networks. This paradigm offers several advantages for deployment: keypoints are compact, more privacy-preserving than storing or processing raw video, and can be computed efficiently on-device. The current state of the art (Joshi et al., 2025) on the iSign benchmark uses a keypoint-based architecture with a standard transformer (Vaswani et al., 2017) operating on two-dimensional joint coordinates. Hybrid methods that combine keypoints with image crops have also been explored (Jang et al., 2026). Our work extends the keypoint-only direction by incorporating additional keypoint channels (depth and visibility) and leveraging a stronger pretrained decoder, while maintaining a focus on computational efficiency and mobile deployability.

3 Methodology

This section presents the architecture and training methodology of ISLiM. We begin by formalizing the SLT problem, then describe the input data and pre-processing steps, followed by the encoder-decoder design. The overall architecture of the proposed framework is shown in Figure 2.

3.1 Problem Definition

SLT, unlike conventional Machine Translation, translates a given Sign Language Video into Spoken Language Text. More formally, given a sequence of T video frames $V = \{frame_1, frame_2, frame_3, \ldots, frame_T\}$, we attempt to generate a spoken-language translation $S = \{word_1, word_2, word_3, \ldots, word_n\}$ in the target language.

3.2 Input Pre-processing

We use Mediapipe Holistic (MediaPipe) to extract 94 upper-body landmarks per frame (see Figure 3 for a comparison with the 76-keypoint representation used by PoseStitch-SLT) (21 per hand, 6 body-pose joints, 46 face points), each with (x, y, z, visibility), forming a 376-dimensional vector. Sequences are normalized per-axis to [0, 1] and padded or sub-sampled to 128 frames.

All reference translations and inferred translations are processed according to the criteria defined in the PoseStitch paper (Joshi et al., 2025) including: lowercasing, contraction expansion, punctuation removal, digit-by-digit number splitting, person name anonymization, and replacing words with fewer than 3 occurrences with <UNK>. Preprocessing is applied to all input references and outputs generated by the model to maintain uniformity when comparing. The rare-word dictionary is computed on the training split only and applied, frozen, to the validation and test references.

3.3 Encoder

The encoder transforms keypoint sequences into contextualized representations in two stages. The visual encoder focuses purely on pose feature extraction

Temporal Convolution Each frame's 376-dimensional vector is passed through two stacked 1D convolutional layers (kernel size 5 along the time axis), reducing dimensionality to 256. For mean-duration videos (∼5 seconds at 25 fps, subsampled to 128 frames), the kernel spans approximately 200 ms of signing; for longer videos the effective span is wider due to subsampling. These layers capture short-range motion patterns, while longer-range temporal dependencies are handled by the transformer encoder that follows.

Self-Attention Transformer The convolution output is projected to 512 dimensions, combined with learned positional embeddings, and processed by a 4-layer transformer encoder. Self-attention enables the non-monotonic SOV-to-SVO alignment without imposing a strict ordering prior on the encoder.

3.4 Decoder

We fine-tune a pretrained T5-small (Raffel et al., 2020) model as the decoder. We pass the full 128-frame representation sequence into the T5 model, which conditions on the temporally aligned frames through cross-attention rather than compressing them into a single vector. This design offloads the temporal alignment and translation burden to the pretrained decoder. For cross-lingual transfer to German, we initialize from the iSign-trained checkpoint, swap the decoder for a German T5 (S et al., 2022) and fine-tune on Phoenix-2014-T.

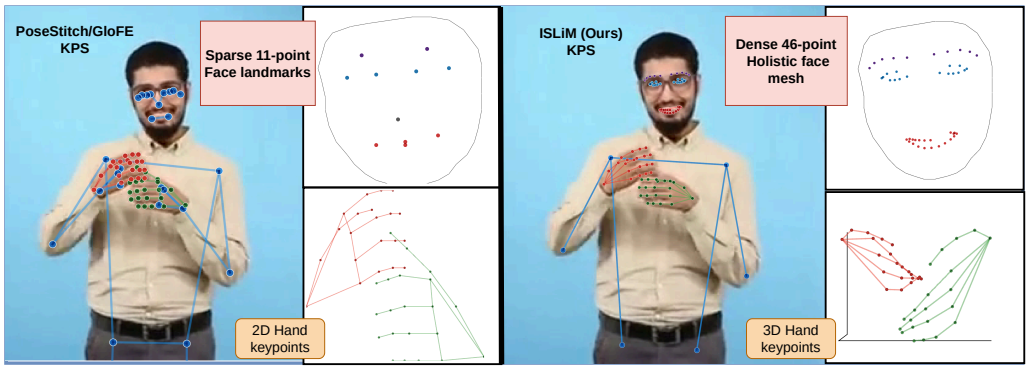


Figure 3: **Keypoint comparison of PoseStitch-SLT (76×2D) and ISLiM (94×4D).** Left: 76 landmarks (33 full-body pose incl. legs/feet (including **11 face**), 42 hands, 1 face reference), x,y only, no depth or visibility. Right: 94 landmarks (6 upper-body pose, 42 hands, **46 face mesh points** capturing lips, eyes, and eyebrows), plus estimated z-depth and per-landmark visibility scores. Same signing frame shown on both sides.

3.5 Multi-Loss Training Strategy

On subsets of iSign smaller than approximately 15K samples, CE-only training regularly collapses: the decoder converges to outputting one or two high-frequency tokens for every input, and once this happens, the gradient signal from CE is too weak to recover. We address this with three auxiliary losses, each mitigating a different aspect of the collapse. A **CLIP contrastive loss** aligns pose and text embeddings in a shared latent space, forcing the encoder to produce distinct representations for different inputs rather than collapsing to a single point. A **Semantic Similarity Loss** compares the decoder's hidden state to a Sentence-BERT (Reimers and Gurevych, 2019; Hamidullah et al., 2024) reference embedding, providing a gradient signal even when token-level overlap (Papineni et al., 2002) is zero. **Entropy Regularization** penalizes the output distribution when it becomes too peaked too early, keeping the decoder in an exploratory regime for longer. Detailed formulations, weights, and ramping schedules are provided in Section 4.2.

4 Implementation Details

4.1 Dataset

The iSign dataset (Joshi et al., 2024) contains 127,237 parallel ISL-to-English samples collected from Deaf signers and interpreters across India, covering news, daily conversation, and educational domains. We apply the PoseStitch (Joshi et al., 2025) text preprocessing pipeline to all references and create our own 90/5/5 random split (seed = 42) as the PoseStitch data split is not available. The split is per-clip, as iSign provides no signer-level metadata; the sample identifiers, which encode each clip's source video, are released with the code.

The iSign 10K subset, introduced in SpaMo-OF (Chowdhury and Sanyal, 2025), is a curated 10,000-sample subset used for controlled small-data experiments. Phoenix-2014-T provides German weather-forecast translations used for cross-lingual transfer experiments; its constrained vocabulary makes it a dramatically easier benchmark, as discussed in Section 5.2.

4.2 Training Objectives and Hyperparameters

Primary objective : Cross-Entropy (CE) Standard next-token prediction with label smoothing ($\epsilon = 0.1$). This is the sole training objective when sufficient data is available (full iSign, 113K training samples).

Multi-loss Configuration Each auxiliary loss contributes to the combined objective.

The **CLIP contrastive loss** aligns mean-pooled encoder features $V \in \mathbb{R}^{B\times d}$ with mean-pooled decoder text embeddings $T \in \mathbb{R}^{B\times d}$:

$$S = (V \cdot T^{\top}) \cdot \tau,$$

$$\mathcal{L}_{\text{CLIP}} = -\frac{1}{2}\Big(\sum_i \log \frac{e^{S_{ii}}}{\sum_j e^{S_{ij}}} + \sum_i \log \frac{e^{S_{ii}}}{\sum_j e^{S_{ji}}}\Big)$$

where $\tau = \exp(2.6592) \approx 14.3$ is a learnable temperature. The computation is performed in `float32` to prevent an observed `float16` drift in the similarity matrix.

The **Semantic Similarity loss** guides the decoder output toward the reference in Sentence-BERT embedding space:

$$\mathcal{L}_{\text{sem}} = 1 - \cos\big(\,\text{proj}(\mathbf{h}_{\text{dec}}),\, \mathbf{e}_{\text{ref}}\big)$$

where proj is a learned linear layer $512 \rightarrow 384$ followed by ℓ_2 normalization, and $\mathbf{e}_{\text{ref}}$ is a precomputed embedding from `all-MiniLM-L6-v2` (Reimers and Gurevych, 2019). For German Phoenix, the multilingual variant `paraphrase-multilingual-MiniLM-L12-v2` (Reimers and Gurevych, 2020) replaces it.

Entropy regularization penalizes overconfident per-token distributions:

$$H = -\sum_{v} p_v \log p_v,$$

$$\mathcal{L}_{\text{ent}} = \frac{1}{N}\sum_{t=1}^{N} \max(0,\, 3.0 - H_t)^2$$

masked over non-padding positions.

The combined objective is:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + 1.0 \cdot \mathcal{L}_{\text{CLIP}} + 0.05 \cdot \mathcal{L}_{\text{sem}} + 0.01 \cdot \mathcal{L}_{\text{ent}}$$

CLIP and semantic weights ramp linearly from 0 to their final values over the first 10 epochs. The weights are determined empirically: a CLIP weight of 1.0 is required to prevent encoder collapse (lower values 0.1–0.5 are insufficient), while semantic and entropy weights are kept low to serve as regularizers rather than dominant objectives.

Optimization. We use AdamW (Loshchilov and Hutter, 2019) ($\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay $= 0.01$), following standard practice for fine-tuning Transformer-based language models. The learning rate follows a linear warmup to stabilize early training, then decays via a cosine schedule, providing a smooth annealing trajectory without abrupt drops at epoch boundaries. We train with a per-device batch size of 4 and gradient accumulation over 4 steps, yielding an effective batch size of 16; the small per-step allocation keeps peak VRAM below 5 GB, making full-dataset training feasible on a single consumer GPU or a small shared cloud instance. Automatic mixed precision (`float16`) further reduces memory and accelerates tensor-core operations, with the CLIP loss computed in `float32` to avoid an observed stability drift in the similarity matrix. Under this configuration, training on the full 113K iSign samples requires approximately 90 GPU-hours.

4.3 Experimental Setup

iSign full-dataset training. For iSign full-dataset training, the visual encoder is initialised at random and the T5-small from pretrained weights; training proceeds on the 113K-sample PoseStitch-preprocessed split with CE-only loss. Training runs for 142 epochs; we observed convergence by epoch 103, with negligible improvement thereafter. The best checkpoint is selected by validation BLEU4.

Small-data (iSign 10K (from SpaMo-OF)). The full 4-loss objective is trained for 500 epochs under the same initialization scheme.

Cross-lingual transfer (Phoenix-2014-T). The ISLiM checkpoint trained on full iSign is fine-tuned on Phoenix-2014-T (German) for 163 epochs. The T5-small decoder is replaced with `GermanT5-efficient-small`; encoder weights transfer directly.

Keypoint normalization. Each of the three spatial channels (x, y, z) is independently min-max normalized to $[0, 1]$ per video on a per-axis basis. Normalization is applied after keypoint extraction and before padding or temporal subsampling to 128 frames. This per-video normalization removes dependence on signer position, camera distance, and frame resolution.

Evaluation metrics. We report multiple BLEU variants across experiments, each serving a different comparison purpose. Two scoring functions are used: (i) PoseStitch BLEU (PS-B4) (originally from (Camgoz et al., 2018b)), the `compute_bleu()` implementation from PoseStitch with add-1 smoothing and 4-gram precision (words tokenized by `str.split()`), and (ii) SacreBLEU (Post, 2018), the standard corpus-level BLEU with `tokenize='13a'` and exp smoothing. On iSign, the primary metric is PS-B4, matching the PoseStitch-SLT evaluation protocol exactly. On SpaMo-OF, we use SacreBLEU to align with the SpaMo-OF paper. We additionally report BLEU-1 through BLEU-3 (sacrebleu), ROUGE-L (Lin, 2004) (Google `rouge_score` with Porter stemmer), METEOR (Banerjee and Lavie, 2005)(NLTK), chrF++ (sacrebleu, `word_order=2`), semantic similarity (SentenceTransformer `all-MiniLM-L6-v2` cosine).

We find several different BLEU implementations that differ in smoothing (add-1 vs. exp vs. none), tokenization, and aggregation (corpus vs. per-sentence). We discuss the empirical impact of these choices in Section 5.2 alongside the results.

5 Results

5.1 Comparison with Prior Work

Table 1 compares ISLiM against prior work across three SLT benchmarks. GloFE and PoseStitch-SLT report results obtained under their own splits; ISLiM is evaluated on our released 90/5/5 split (seed = 42), with references and outputs processed identically. ISLiM attains a BLEU-4 of **5.85** on iSign, exceeding the published PoseStitch-SLT result (3.43) at a comparable parameter scale (75M vs. 54M); since the protocols differ, the comparison is indicative, and the controlled contribution is the preprocessing ablation, which shows the identical baseline already exceeds the published value (2.24 to 4.10) when evaluated under the PoseStitch pipeline (Section 5.2). The bootstrap 95% confidence interval for sacreBLEU-4 is [5.53, 6.15] (1,000 resamples over test examples), indicating stability of the estimate with respect to test sampling. Training seed variability is not reported. No large-scale visual or video pretraining is employed. Figure 1 plots the parameter-efficiency trade-off. A per-metric breakdown is given in Appendix A.

On the iSign 10K subset (SpaMo-OF), ISLiM's 75M keypoint-based model achieves BLEU-4 of 4.12, below the 8.58 of the 3.6B video-based SpaMo-OF but substantially higher than the SLT(GF) and GFSLT-VLP video baselines presented in the paper (Chowdhury and Sanyal, 2025). For cross-lingual transfer, ISLiM is first trained on full iSign (English ISL), then fine-tuned on Phoenix-2014-T (German DGS) with only the decoder swapped for a German T5. The fine-tuned 75M model achieves BLEU-4 14.99 vs. SpaMo-OF's reported 14.72, performing comparably to the 3.6B SpaMo-OF on Phoenix with a 48$\times$ parameter disadvantage. This suggests that keypoint features capture enough language-agnostic signing structure to transfer across unrelated sign languages. We also note that video-based models are not yet benchmarked on the full iSign dataset; even the SLT (GF) baseline degrades sharply on the full iSign set (BLEU-4 0.32, per SpaMo-OF's results), reflecting the greater difficulty of the broader-domain benchmark.

5.2 Dataset Quality and Metric Reliability

Dataset quality The iSign dataset covers a broad range of topics: topic modeling reveals over 1,700 distinct themes spanning news, sports, education, entertainment, health, and cultural content. It includes content related to Deaf education and accessibility. Like many large-scale sign language datasets, it exhibits a long-tail vocabulary distribution: 48.2% of unique words appear exactly once, and 77.6% appear five or fewer times. With 68,029 unique words across 113K training samples, this distribution reflects the diversity of the domain rather than a data quality issue. For the SLT task, this means that some evaluation references contain vocabulary that is inherently difficult to model from limited occurrences, contributing to the observed 20–21% zero-BLEU rate. This highlights an opportunity for future dataset efforts to consider vocabulary coverage and multiple references per video, which would support more reliable model evaluation.

The iSign-10K subset in SpaMo-OF provides a useful controlled setting for small-data experiments and multi-loss ablation, enabling comparisons that would be computationally prohibitive on the full dataset. We note that its topic distribution skews more towards education and learning relative to the full iSign dataset, and it contains some near-duplicate sentences across train and test splits. These characteristics are worth considering when interpreting cross-dataset comparisons and point toward the value of designing future evaluation subsets with explicit coverage targets and controlled overlap (Alkain et al., 2026).

In general, many large-scale SLT datasets are constructed by chopping long, continuous signing videos into shorter clips. While pragmatic, such segmentation can sever contextual cues. Tanzer et al. (Tanzer et al., 2024) caution against over-reliance on sentence-segmented references as ground truth. We align with this caution and treat sentence-segmented references as a practical, though imperfect, training and evaluation target rather than as a gold standard of signing intent.

Because SLT datasets typically consist of single video-sentence pairs rather than conversational chunks, we find sentence-level BLEU (sent-BLEU) more informative than corpus BLEU for detecting collapse in our setting. We also observe that Semantic Similarity (SemSim) provides a complementary signal by evaluating meaning preservation

Method	KP-dim	KP	Video	BLEU-4	ROUGE-L	Params
iSign (full) Dataset						
GloFE	2D	✓		0.55	9.08	42M
PoseStitch-SLT	2D	✓		3.43	12.50	54M
ISLiM (ours)	4D	✓		**5.85**	**20.12**	**75M**
iSign 10K Subset (SpaMo-OF) Dataset						
SLT (GF)	–		✓	0.73	9.08	∼40M
GFSLT-VLP	–		✓	1.93	9.24	∼680M
SpaMo	–		✓	7.35	16.23	∼3.6B
SpaMo-OF	–		✓	**8.58**	**18.98**	∼3.6B
ISLiM (ours)	4D	✓		4.12	12.66	**75M**
Phoenix-2014-T Dataset						
SpaMo-OF	–		✓	14.72	33.52	∼3.6B
ISLiM (ours)	4D	✓		**14.99**	**35.06**	**75M**

Table 1: ISLiM vs. prior work. Numbers from (Chowdhury and Sanyal, 2025) except iSign GloFE/PoseStitch from (Joshi et al., 2025). KP-dim = keypoint dimensionality; KP = keypoint-based; Video = video-based. ISLiM improves previously reported iSign keypoint SOTA by 71% and matches the 3.6B SpaMo-OF on Phoenix-14T at 48× fewer parameters. Bootstrap 95% CIs (1,000 resamples): iSign sacreBLEU-4 5.84 [5.53, 6.15].

independently of exact n-gram overlap, particularly valuable in our setting where hapax words or synonym choice can inflate zero-BLEU rates. However, SemSim has its own weakness: risk of moderate similarity to generic English sentences. In our experiments, combining sent-BLEU and SemSim formed a practical evaluation kit: sent-BLEU captures lexical overlap, while SemSim captures semantic adequacy.

Mobile feasibility. Preliminary deployment on a standard Android phone shows promising results: the model achieves translation quality comparable to desktop beam-search decoding with 384 ms mean inference latency (see Appendix D for details).

6 Conclusion

We present ISLiM, a compact keypoint-based gloss-free SLT architecture that achieves state-of-the-art results among reported keypoint-only models on iSign. Our findings underscore three practical lessons: careful text preprocessing can dominate reported BLEU scores and must be rigorously controlled for fair comparison; pretrained language models supply syntactic fluency that would be impractical to learn from sign-language data alone; and lightweight keypoint-based models can run on consumer smartphones at acceptable latency. Cross-lingual transfer to German Sign Language on Phoenix-2014-T further demonstrates that a shared visual encoder paired with language-specific decoders is a viable path toward multilingual SLT.

6.1 Future Work

Key directions include: (i) new ISL datasets with controlled vocabulary coverage and multiple references per video; (ii) SLT-specific evaluation protocols that account for high hapax rates and single-sentence settings; (iii) encoder pretraining on large-scale keypoint corpora from other sign languages or general body-motion data; and (iv) identifying robust metric settings or a combination of metrics that capture lexical overlap, semantic adequacy, and fluency in a single reliable evaluation framework for SLT.

7 Limitations

- **Single-reference evaluation.** iSign provides one reference per video; multi-reference evaluation would allow more reliable assessment when multiple valid translations exist.
- **No human evaluation.** All improvements are measured by automatic metrics, which may over-reward fluent but factually incorrect output. Human judgment is needed to confirm that measured gains correspond to genuine quality improvements.
- **Keypoint fidelity.** MediaPipe Holistic may

lose some non-manual markers that carry grammatical information in ISL.

- **Failure mode taxonomy.** Zero-BLEU predictions include a mix of collapse and paraphrases; a quantitative breakdown is left to future work.

8 Ethical Considerations

- **Data consent.** The iSign dataset is collected from ISL signers and interpreters in India. We rely on the dataset creators' assurances regarding consent.

- **Privacy.** Our model uses MediaPipe keypoints rather than raw video, discarding identifiable pixel content. Keypoint sequences may nevertheless retain identifying information such as body proportions, facial geometry, and signing style, and no formal privacy guarantee is claimed.

- **Intended use.** ISLiM is a research tool, not a replacement for human interpreters.

- **Dialect coverage.** ISL has regional variation across India. The iSign dataset may not represent all dialects equally.

References

Samuel Albanie, Gül Varol, Liliane Momeni, Hannah Bull, Triantafyllos Afouras, Himel Chowdhury, Neil Fox, Bencie Woll, Rob Cooper, Andrew McParland, and Andrew Zisserman. 2021. BOBSL: BBC-Oxford British Sign Language Dataset. *arXiv*.

Bittor Alkain, Adrián Núñez-Marcos, Carlos Escolano, Laura Docío-Fernández, Olatz Perez-de Viñaspre, and Gorka Labaka. 2026. Critical analysis of datasets for sign language translation. *Frontiers in Artificial Intelligence*, Volume 9 - 2026.

Patricia Cabot Alvarez, X. Nieto, and Laia Tarres Benet. 2022. Sign language translation based on transformers for the how2sign dataset.

Sobhan Asasi, Mohamed Ilyes Lakhal, and Richard Bowden. 2025. Hierarchical feature alignment for gloss-free sign language translation. In *Adjunct Proceedings of the 25th ACM International Conference on Intelligent Virtual Agents*, IVA Adjunct '25, New York, NY, USA. Association for Computing Machinery.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018a. Neural sign language translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018b. Neural sign language translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7784–7793.

Necati Cihan Camgöz, Oscar Koller, Simon Hadfield, and R. Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10020–10030.

Yutong Chen, Fangyun Wei, Xiao Sun, Zhirong Wu, and Stephen Lin. 2022. A simple multi-modality transfer learning baseline for sign language translation. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5110–5120.

Anal Roy Chowdhury and Debarshi Kumar Sanyal. 2025. Enhancing Indian Sign Language translation via motion-aware modeling. In *Proceedings of the Workshop on Sign Language Processing (WSLP)*, pages 39–46, IIT Bombay, Mumbai, India (Co-located with IJCNLP–AACL 2025). Association for Computational Linguistics.

Emanuele Colonna, Ivan Rinaldi, David Landi, Gennaro Vessio, and Giovanna Castellano. 2026. Handscribe: A gloss-free framework for sign language translation and gloss sequence generation. *Computer Vision and Image Understanding*, 265:104674.

Mathieu De Coster and Joni Dambre. 2022. Leveraging frozen pretrained written language models for neural sign language translation. *Information*, 13(5).

Nguyen Son Dinh, Tuan Dung Nguyen, Duc Tri Tran, Nguyen Dang Huy Pham, Thuan Hieu Tran, Ngoc Anh Tong, Quang Huy Hoang, and Phi Le Nguyen. 2025. Sign language recognition: A large-scale multi-view dataset and comprehensive evaluation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 7887–7897.

Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro-i Nieto. 2021. How2sign: A large-scale multimodal dataset for continuous american sign language. In *2021 IEEE/CVF Conference*

on Computer Vision and Pattern Recognition (CVPR), pages 2734–2743.

Abhigyan Ghosh and Radhika Mamidi. 2022. English to Indian Sign Language:rule-based translation system along with multi-word expressions and synonym substitution. In *Proceedings of the 19th International Conference on Natural Language Processing (ICON)*, pages 123–127, New Delhi, India. Association for Computational Linguistics.

Jia Gong, Lin Geng Foo, Yixuan He, Hossein Rahmani, and Jun Liu. 2024. LLMs are Good Sign Language Translators . In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18362–18372, Los Alamitos, CA, USA. IEEE Computer Society.

Yasser Hamidullah, Josef van Genabith, and Cristina España-Bonet. 2024. Sign language translation with sentence embedding supervision. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 425–434, Bangkok, Thailand. Association for Computational Linguistics.

Eui Jun Hwang, Sukmin Cho, Junmyeong Lee, and Jong C. Park. 2025. An efficient gloss-free sign language translation using spatial configurations and motion dynamics with LLMs. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3901–3920, Albuquerque, New Mexico. Association for Computational Linguistics.

ISLRTC. 2024a. Annual report 23-24 | indian sign language research and training center ,government of india | india.

ISLRTC. 2024b. Isl interpreter list | indian sign language research and training center ,government of india | india.

Youngjoon Jang, Liliane Momeni, Zifan Jiang, Joon Son Chung, Gül Varol, and Andrew Zisserman. 2026. Lost in translation, found in embeddings: Sign language translation and alignment. *arXiv*.

P Jayachandran*. 2026. Indian sign language: A natural, fully developed language of global stature and cultural depth.

Zifan Jiang, Gerard Sant, Amit Moryossef, Mathias Müller, Rico Sennrich, and Sarah Ebling. 2024. SignCLIP: Connecting text and sign language by contrastive learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 9171–9193, Miami, Florida, USA. Association for Computational Linguistics.

Abhinav Joshi, Romit Mohanty, Mounika Kanakanti, Andesha Mangla, Sudeep Choudhary, Monali Barbate, and Ashutosh Modi. 2024. iSign: A benchmark for Indian Sign Language processing. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10827–10844, Bangkok, Thailand. Association for Computational Linguistics.

Abhinav Joshi, Vaibhav Sharma, Sanjeet Singh, and Ashutosh Modi. 2025. PoseStitch-SLT: Linguistically inspired pose-stitching for end-to-end sign language translation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 13834–13853, Suzhou, China. Association for Computational Linguistics.

Shruthi Koratagere Anantha Kumar, Gangavarapu Vigneswara Ihita, Sachin Chaudhari, and Paventhan Arumugam. 2022. A survey on rural internet connectivity in india. In *2022 14th International Conference on COMmunication Systems and NETworkS (COMSNETS)*, pages 911–916.

Colin Lea, Michael D. Flynn, René Vidal, Austin Reiter, and Gregory D. Hager. 2017. Temporal convolutional networks for action segmentation and detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1003–1012.

Peike Li, Trevor Cohn, and Jianyuan Guo. 2025. Bridging sign and spoken languages: Pseudo glossgeneration for sign language translation. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Annual Meeting of the Association for Computational Linguistics*.

Kezhou Lin, Xiaohan Wang, Linchao Zhu, Ke Sun, Bang Zhang, and Yi Yang. 2023. Gloss-free end-to-end sign language translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12904–12916, Toronto, Canada. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Jiannan Mao, Chenchen Ding, Tadahiro Matsumoto, Hideki Tanaka, and Masao Utiyama. 2026. Conditional variational autoencoder for gloss-free sign language translation. In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 17317–17321.

Google MediaPipe. MediaPipe Holistic — Simultaneous Face, Hand and Pose Prediction, on Device — research.google. `https://research.google/blog/mediapipe-holistic-simultaneous-face-hand-and-pose-prediction-on-device/`. [Accessed 03-07-2026].

Donna Jo Napoli and Rachel Sutton-Spence. 2014. Order of the major constituents in sign languages: implications for all language. *Frontiers in Psychology*, 5.

Yiming Ni, Zhi-Qi Cheng, Jiayu Li, and Wei Cheng. 2026. Sign-language datasets at scale: A comprehensive survey on resources, benchmarks, and annotation standards. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 41578–41604, San Diego, California, United States. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *Preprint*, arXiv:2103.00020.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Schweter S, Philip May, and Philipp Schmid. 2022. Germant5/t5-efficient-gc4-all-german-small-el32.

Bowen Shi, Diane Brentari, Gregory Shakhnarovich, and Karen Livescu. 2022. Open-domain sign language translation learned from online video. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6365–6379, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

S. Sinha. 2018. *Indian Sign Language: A Linguistic Analysis of Its Grammar*. Gallaudet University Press.

Sugandhi, Parteek Kumar, and Sanmeet Kaur. 2020. Sign language generation system based on indian sign language grammar. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 19(4).

Garrett Tanzer, Maximus Shengelia, Ken Harrenstien, and David Uthus. 2024. Reconsidering sentence-level sign language translation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6262–6287, Miami, Florida, USA. Association for Computational Linguistics.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. 2022. VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Advances in Neural Information Processing Systems*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

WHO. 2026. Who fact sheet | deafness and hearing loss.

Ryan Wong, Necati Cihan Camgoz, and Richard Bowden. 2024. Sign2GPT: Leveraging large language models for gloss-free sign language translation. In *The Twelfth International Conference on Learning Representations*.

Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press.

Jinhui Ye, Xing Wang, Wenxiang Jiao, Junwei Liang, and Hui Xiong. 2024. Improving gloss-free sign language translation by reducing representation density. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Kayo Yin and Jesse Read. 2020. Better sign language translation with STMC-transformer. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5975–5989, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Benjia Zhou, Zhigang Chen, Albert Clapés, Jun Wan, Yanyan Liang, Sergio Escalera, Zhen Lei, and Du Zhang. 2023. Gloss-free sign language translation: Improving from visual-language pretraining. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20871–20881.

Hao Zhou, Wengang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. 2021. Improving Sign Language Translation with Monolingual Data by Sign Back-Translation . In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1316–1325, Los Alamitos, CA, USA. IEEE Computer Society.

A Full ISLiM Metric Matrix on iSign

Metric	Val	Test
Corpus BLEU (sacrebleu, `tokenize='13a'`*)*		
BLEU-1	22.53	22.33
BLEU-4	6.19	5.84
Sentence BLEU (sacrebleu, `effective_order=True`*)*		
BLEU-1 (sent avg)	18.95	18.86
BLEU-4 (sent avg)	7.71	7.76
PoseStitch BLEU (add-1 smoothing, `str.split()`*)*		
PS-B1	22.54	22.33
PS-B4	6.19	5.85
Other metrics		
ROUGE-L	20.28	20.12
METEOR	16.91	16.72
chrF++	20.44	20.48
Semantic similarity	33.69	32.92
Sent zero-BLEU rate	20.3%	21.3%

Table 2: Comprehensive ISLiM metrics on iSign (val 6,302 / test 6,304).

B Qualitative Examples

Table 3 presents representative validation-set translations illustrating recurring success and failure modes.

C Input Channel Sensitivity

We evaluate ISLiM on the iSign test set with keypoint channels zeroed at inference time on a model trained with the full input, shown in Table 4. Removing z-depth causes a modest drop, while zeroing visibility or face landmarks collapses the model.

D Mobile Deployment Feasibility Study

Model weights are exported from the training checkpoint (`best_ps.pt`) using PyTorch's mobile optimization module and converted to TorchScript format via the `torch.jit.trace` API. The traced modules were then serialized using the `optimize_for_mobile` utility, which applies operator fusion, constant folding, and dead-code elimination for efficient on-device execution. Six modules were traced:

The test device is a Google Pixel 6a (Google Tensor, 6 GB RAM, Android 16). Models are loaded via `LiteModuleLoader` and inference is performed on CPU with greedy decoding (argmax per step, max 64 tokens).

Inference latency and quality. Table 6 summarises the on-device evaluation over 57 keypoint-based test samples: mean latency is 398 ms (min 208, max 789). The mobile greedy decode slightly exceeds the desktop beam-search PS-BLEU-4 (6.77 vs. 5.85), which is within sampling variance due to the restricted subset size. Latency covers neural-network inference only (pose encoder and greedy T5 decoding); camera acquisition and on-device keypoint extraction are excluded.

Memory: All decoder, embedding, and lm-head modules are resident simultaneously during decoding (on-device model size in Table 6). Under `android:largeHeap="true"`, the app completes the subset evaluation and does not crash until after the evaluation completed.

E SacreBLEU Signatures

All corpus and sentence-level scores are computed with sacreBLEU v2.4.2. The canonical signatures for each metric configuration are listed below for reproducibility.

- **Corpus BLEU** (`BLEU` corpus_score, tokenize='13a', exp smoothing):
  `nrefs:1|case:lc|eff:no|tok:13a|smooth:exp|version:2.4.2`
- **Sentence BLEU** (`BLEU` sentence_score with effective_order=True):
  `nrefs:1|case:lc|eff:yes|tok:13a|smooth:exp|version:2.4.2`
- **chrF++** (`CHRF` corpus_score, word_order=2):
  `nrefs:1|case:lc|eff:yes|nc:6|nw:2|space:no|version:2.4.2`

The PoseStitch BLEU (PS-BLEU) score is computed with a separate implementation (`compute_bleu()` with add-1 smoothing and `str.split()` tokenization) and does not have a sacreBLEU signature.

F Multi-Loss Training Dynamics

Figure 4 plots validation BLEU-4 across training for the SpaMo-OF 10K subset, comparing CE-only training against the full 4-loss configuration. CE-only training either collapses to near-zero (BLEU-4 < 0.3 after 24 epochs) or plateaus at $\sim$3 even after 325 epochs. The 4-loss configuration climbs steadily past 5, reaching a maximum validation BLEU-4 of 5.44 at epoch 500 before mild overfitting. Validation BLEU-4 uses sentence-averaged

Case	Reference	ISLiM output
(a) Clean	there were 6 8 passengers and 4 crew members onboard in all there were 7 2 people	there were 6 6 passengers and 4 crew members onboard the flight
(b) Clean (almost)	sushant singh rajput s father krishna kishore singh has filed a case in the delhi high court	sushant singh rajput s parents filed a case in the delhi high court
(c) Paraphrase	they said the plane was at a very high speed which is why it went off the runway	this is because of the aircraft s failure on the runway
(d) Collapse	their internet is very slow	the person person
(e) Halluc.	his family has said once the covid outbreak ends	covid 1 9 has spread all over the world

Table 3: Qualitative validation-set examples. (a–b) Clean predictions preserving numerical and entity detail. (c) Partial semantic match with low BLEU. (d) Collapse on a short reference with hapax vocabulary. (e) Topical hallucination that semantic-similarity metrics over-reward.

Condition	BLEU-4	Δ	Zero%
Full (94×4D)	5.84	—	21.3
− z-depth	5.33	–0.51	22.9
− visibility	2.21	–3.63	33.1
− face (46 kp)	0.37	–5.47	37.7
Hand-only (42 kp)	0.21	–5.63	36.0

Table 4: Inference-time input-channel sensitivity (6,304 test samples). Zeroing visibility or face landmarks collapses the model. As channels are removed from a fully trained model, these results indicate sensitivity to feature removal under distribution shift rather than the isolated contribution of each channel; retraining-based ablations are left to future work

Module	Size
`encoder_traced.ptl`	56 MB
`t5_encoder.ptl`	73 MB
`t5_decoder_first_step.ptl`	159 MB
`t5_decoder_step.ptl`	147 MB
`t5_shared_embed.ptl`	63 MB
`t5_lm_head.ptl`	63 MB
Total	**561 MB**

Table 5: Sizes of the exported TorchScript artifacts (`.ptl` files) for mobile deployment; the on-device model footprint is reported in Table 6.

scoring; the corresponding corpus-level test set BLEU-4 at the selected checkpoint is 4.12 (Table 1).

G Parameter Efficiency on Phoenix-2014-T

Figure 5 shows the parameter efficiency of ISLiM, SpaMo, and SpaMo-OF on the Phoenix-14-T dataset.

Quantity	Value
Evaluation samples	57
Model size (on-device)	495.7 MB
Peak RAM (heap)	38.8 MB
CPU cores	8
Latency (mean)	398 ms
Latency range (min–max)	208–789 ms
PS-BLEU-1/2/3	24.3 / 13.81 / 9.19
PS-BLEU-4	6.77

Table 6: On-device evaluation of the keypoint-to-English pipeline (57 iSign test samples, Google Pixel 6a, CPU-only greedy decoding).

H Qualitative Analysis of Multi-Loss Setup

Table 7 summarizes, qualitatively, how each auxiliary loss contributed in our SpaMo-OF 10K training.

Loss	Prevents Collapse	Improves BLEU	Improves Fluency
CE only	No	Baseline	Good (when stable)
CE + CLIP	Yes	Strong	Moderate
CE + Semantic	Partial	Moderate	Strong
CE + Entropy	Yes	Slight	No

Table 7: Qualitative contribution of each auxiliary loss. CLIP contrastive loss is the primary driver of BLEU improvement; semantic similarity and entropy regularization act as stabilizers.

I Effect of Sentence Length on Translation Quality

We examine whether translation quality degrades with increasing sentence length. Table 8 reports mean and median BLEU-4 and ROUGE-L on the full iSign test set (6,304 sentences), binned by reference word count.

Sentence length has a negligible effect on translation quality. Mean BLEU-4 remains stable at

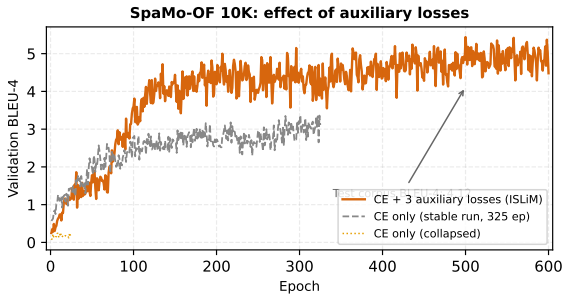


Figure 4: Validation BLEU-4 on iSign 10K. CE-only runs either collapse (orange dotted, stuck <0.3) or plateau at $\sim$3 (grey dashed, 325 ep). The 4-loss configuration (orange solid, 600 ep) reaches 5.44, demonstrating that auxiliary losses, not architecture, are the source of the gain.

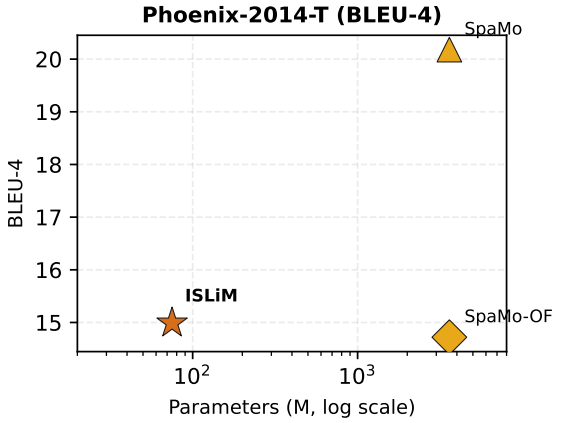


Figure 5: Parameter efficiency on Phoenix-2014-T. ISLiM (75M, red star) matches the 3.6B SpaMo-OF at 48$\times$ fewer parameters (BLEU-4 14.99 vs. 14.72). Keypoint-based ISLiM appears as a star; video-based methods as triangles/diamonds.

approximately 7 across the 5-25 word range that covers 93% of the test set, dropping only to 5.6 for the small number of sentences exceeding 25 words ($n = 46$). Mean ROUGE-L increases slightly with length (16.4 to 22.5), indicating that surface-level word overlap remains consistent even when higher-order n-gram matches become more challenging.

The only notable pattern is in very short sentences (1–4 words): this bucket has the highest mean BLEU (13.84) but a median of zero, meaning more than half of these sentences score no n-gram matches at all while the remainder score high. This bimodality reflects the reduced redundancy of short targets, suggesting that there is no margin for partial credit, so the model either matches the reference exactly or receives no n-gram overlap.

Predicted sentence length slightly lags the reference length as sentences grow longer (e.g., 27.2 vs. 16.6 words for the 26–30 word bucket), a common length bias in neural machine translation. However, sacreBLEU's brevity penalty largely compensates for this effect, as evidenced by the flat BLEU-4 trend. Overall, the model's translation quality is independent of sentence length for the vast majority of test examples.

Words	#Sent.	Mean B4	Median B4	Mean RL	Pred. Len
1–4	662	13.84	0.00	18.28	5.2
5–7	1021	7.21	4.20	16.42	7.8
8–10	1323	7.26	4.46	19.32	9.9
11–13	1211	7.31	4.07	20.63	11.7
14–16	974	6.99	3.65	21.02	13.0
17–20	734	6.76	3.60	21.20	14.6
21–25	324	6.02	3.89	21.00	16.0
26–30	46	5.61	3.81	22.46	16.6

Table 8: BLEU-4 and ROUGE-L by reference sentence length on the full iSign test set. Mean BLEU-4 is stable at approximately 7 for sentences of 5–25 words, with only a mild drop for longer sentences. Short sentences (1–4 words) show high variance: mean 13.84 but median 0.00, indicating a bimodal distribution where the model either translates them correctly or fails entirely.

Whitening and Fine-Tuning Are Substitutes: One-Shot Indian Sign Language Recognition

Galiveeti Poornima
Dept. of Engineering and Technology
S-VYASA Deemed to be University
Bengaluru, India
varaprabha.lita@gmail.com

Sandeep Mathias
Dept. of Computer Science and Engg.
Presidency University
Bengaluru, India
sandeepalbert@presidencyuniversity.in

Abstract

We describe our submission to the WSLP 2026 shared task on Isolated Sign Language Recognition. The task is one-shot retrieval over 4,361 Indian Sign Language word classes with roughly one training clip per class. Our final system placed 3rd on top-1 accuracy at 0.6825 and 2nd on top-10 at 0.7357. The system is deliberately simple. It combines MediaPipe keypoints, a graph attention backbone pretrained on a different ISL corpus, PCA whitening, and nearest-exemplar cosine retrieval against an enriched gallery. Our main finding concerns how two very differently priced interventions relate. PCA whitening of the frozen cross-corpus encoder is worth $+18.85$ points on a leave-one-out gallery evaluation. Fine-tuning that encoder end-to-end is worth a comparable amount. The two do not compose. Once the encoder has been fine-tuned, whitening adds $+0.38$ points, an interaction of -18.5. Whitening and fine-tuning act as substitutes because both correct the same embedding anisotropy, and a transform that is fitted in seconds captures most of what a day of GPU training buys. We also report fifteen interventions that produced no usable gain: twelve indistinguishable from zero and three actively harmful. These were evaluated with McNemar tests on a 2,030-query held-out split with a minimum detectable effect of 1.23 points. We argue that the 526-clip official test set is too small to support the comparisons participants routinely make with it.

1 Introduction

Indian Sign Language (ISL) serves a large signing population. It is supported by a small fraction of the annotated data available for the sign languages that drive most published progress. The WSLP 2026 Isolated Sign Language Recognition shared task confronts this shortage directly (WSLP 2026 Shared Task Organisers, 2026). Participants receive roughly 4,398 training clips spanning about 4,361 word classes. That is approximately one example per class.

This is not a low-data variant of a standard classification problem. At a mean of 1.01 clips per class, it is a one-shot retrieval problem. Any method whose gains depend on gradient signal per class is structurally disadvantaged.

Our approach was shaped by an early observation. We applied a graph attention backbone (Patra et al., 2024), pretrained on FDMSE-ISL, to WSLP data without adaptation. Under plain cosine retrieval it scored 0.2510 top-1. This falls far below what the encoder's in-domain performance would predict. The natural diagnosis is that the features are out of domain and need fine-tuning. That diagnosis turned out to be substantially wrong. The features were largely intact. Their geometry was not. A few high-variance directions dominated every cosine comparison. A single closed-form whitening transform moved the system from 0.2510 to 0.6198 on the official test split without touching a model weight.

This paper reports that result and what followed from it. We make five contributions.

1. A shared task system reaching 3rd place at 0.6825 top-1 and 2nd on top-10, built from four components and no ensemble of convenience.
2. An ablation isolating the whitening gain to the variance term rather than to the removal of nuisance axes. Whitening strength moves accuracy by tens of points, while the number of discarded leading components is nearly irrelevant at the operating point.
3. A 2×2 analysis showing that whitening and fine-tuning act as substitutes. Whitening is worth $+18.85$ points on the pretrained encoder and $+0.38$ on the fine-tuned one, an interaction of -18.5.

Proceedings of the Second Workshop on Sign Language Processing, pages 32–42
October 29, 2026 ©2026 Association for Computational Linguistics

4. Evidence that the graph attention backbone depends architecturally on arm geometry. Masking the input to hands alone collapses its accuracy, where an equivalently trained temporal convolutional network is essentially unaffected.
5. Fifteen negative results at stated statistical power, together with an argument that the official test set is too small to adjudicate them.

2 Task and Data

2.1 Task definition

Given a query video of a single isolated sign, a system predicts the word class. Submissions provide a ranked list of ten candidates per query. Scoring is by top-1 accuracy, with top-5 and top-10 also reported.

2.2 Splits

The organisers provide approximately 4,398 training clips over 4,361 classes, 109 validation clips, and 526 test clips. The validation split is far too small to gate decisions. A single clip is worth 0.92 points. We did not use it for model selection.

2.3 Gallery enrichment

The task is retrieval rather than classification. Additional labelled clips can therefore be added to the gallery at inference time without retraining. With organiser sanction we added clips from CISLR (Joshi et al., 2022). These contributed 3,758 additional prototypes, taking the gallery from 4,398 to 8,156 clips. We verified that no `uid` overlapped between the added clips and either the validation or the test split. Even after enrichment, 53.4% of classes remain singletons.

3 Related Work

Indian Sign Language has seen steady growth in corpora but remains thin relative to ASL or DGS. INCLUDE (Sridhar et al., 2020) covers a small vocabulary at high depth. CISLR (Joshi et al., 2022) covers roughly 4,700 words at approximately 1.5 videos per word, which makes it explicitly a one-shot resource. FDMSE-ISL (Patra et al., 2024) inverts the trade-off with 2,002 words recorded by 20 signers. iSign (Joshi et al., 2024) assembles a broader benchmark suite. Closest to our setting, Vandendriessche et al. (2025) treat one-shot ISLR as a retrieval problem and argue that sign-level representations are what make such systems functional, which is the framing we adopt here. The WSLP 2026 task sits at the CISLR end of this spectrum. The methods that succeed there differ accordingly from those developed on depth-rich corpora such as WLASL (Li et al., 2020) or PHOENIX14T (Camgoz et al., 2018).

Skeleton-based recognition avoids the appearance overfitting of RGB pipelines and is the dominant approach for low-resource sign languages. HWGAT (Patra et al., 2024) applies windowed graph attention over an upper-body skeleton graph and reports gains over prior keypoint models across several corpora. Temporal convolutional networks (Bai et al., 2018) remain a strong and considerably simpler baseline. Our fusion results reinforce that point.

The degradation of cosine similarity under uneven embedding variance is well established outside sign language work. PCA whitening is standard in image retrieval (Jégou and Chum, 2012). Removing dominant directions improves word vectors (Mu and Viswanath, 2018). Contextual encoders are known to occupy narrow cones (Ethayarajh, 2019; Su et al., 2021). Our contribution is not the technique. It is the magnitude of the effect in the cross-corpus one-shot setting, and the finding that whitening substitutes for fine-tuning rather than compounding with it.

4 System

4.1 Keypoint extraction

We extract pose, hand and face landmarks with MediaPipe Holistic (Lugaresi et al., 2019), using `model_complexity=2` and detection and tracking thresholds of 0.1. We retain the 29 keypoints used by the backbone's transform chain and resample every clip to 192 frames.

Two practical details cost us time. Detection thresholds of 0.0 produce a degenerate matrix and crash the legacy `mp.solutions` API, so 0.1 is the safe floor. We also pin MediaPipe to version 0.10.14, because landmark conventions are not stable across releases.

The cached tensor layout is $(N, 192, 64, 2)$. This is four windows of 16 keypoints with duplication, not 29 independent keypoints. The four windows are head with right arm and right hand, head with left arm and left hand, head with right arm and left hand, and head with left arm and right hand. Horizontal flip augmentation therefore requires swap-

ping the first two windows, swapping the second two, and swapping the eyes. Negating the x coordinate alone yields anatomically invalid poses. We flag this because the failure is silent. Training proceeds and loss decreases. We verified our permutation empirically, as described in Section 10.

4.2 Backbone and fine-tuning

We use HWGAT (Patra et al., 2024), a hierarchical window-based graph attention network pretrained on FDMSE-ISL. We fine-tune end-to-end with an ArcFace objective (Deng et al., 2019) for 12 epochs at batch size 8 with gradient accumulation 8 and horizontal flip probability 0.5. Training ran on a single RTX 5060, with peak memory of 7.67 GB out of 8.5 GB and 5 to 8 minutes per epoch.

One methodological point deserves emphasis. ArcFace loss fell steeply during fine-tuning, from 6.87 to 2.14, while held-out retrieval accuracy moved by fractions of a point. At approximately 1.87 clips per class after enrichment, the margin objective is easy to satisfy in ways that do not improve retrieval. Loss is not a usable proxy for retrieval quality at this data scale. Every promotion decision in this work was gated on held-out top-1 alone.

4.3 PCA whitening

Let $X \in \mathbb{R}^{N \times D}$ be the gallery embeddings. We compute the mean μ and the principal directions V of the centred matrix by SVD. We drop the leading d components, retain the next k, and rescale each retained axis by $\lambda_i^{-\alpha/2}$, where λ_i is the variance of that axis and α is the whitening strength. The transform $z = (x-\mu)W$, followed by L2 normalisation, is applied to both gallery and query embeddings. The construction follows standard practice in image retrieval (Jégou and Chum, 2012) and sentence representation learning (Su et al., 2021).

Two properties matter for the analysis that follows. First, the fit is computed on the gallery side only and then frozen. We never refit on test queries. Second, α is a continuous knob. At $\alpha = 0$ the transform is a pure rotation and leaves cosine similarity unchanged. At $\alpha = 1$ every retained axis is mapped to unit variance. This lets us attribute the gain in Section 7.

4.4 Retrieval

We score a query against every gallery clip by cosine similarity. For each word class we then take the maximum similarity over that class's prototypes, rather than the similarity to their centroid. This is nearest-exemplar matching rather than prototype matching (Snell et al., 2017). Class-max outperformed centroid averaging consistently.

The interpretation is straightforward. Prototypes for a class come from different signers with different execution styles. Averaging heterogeneous exemplars produces a centroid that resembles none of them. Under one-shot conditions with multi-signer galleries, the nearest exemplar is the right query.

4.5 Score fusion

Our final submission fuses the fine-tuned HWGAT with a temporal convolutional network (Bai et al., 2018) trained from scratch on the same keypoints. Fusion is by z-normalised score averaging at weight 0.6. Section 9 explains why this addition moved the score when six others did not, and what that movement is worth.

5 Evaluation Protocol

The official test set contains 526 clips. Throughout, we write accuracies as percentages and differences between them as points. One clip is worth 0.19 points. The binomial standard error at $p = 0.68$ is approximately 2.03 points. Paired comparisons are more sensitive than that marginal figure suggests, but even under McNemar a difference below roughly 2.4 points is not detectable at $n = 526$ for systems as similar as ours. Many of the interventions we wanted to test have true effects well inside that band. Gating on the leaderboard would have meant promoting noise.

We therefore constructed a held-out split from the enriched gallery. Every class with at least two gallery clips donates one clip to a held-out query set. This yields 2,030 queries against a remaining 6,126-clip gallery. Whitening is fitted on the gallery side only, mirroring the submission pipeline exactly.

We compare systems with McNemar's test (McNemar, 1947) on paired per-query outcomes. This is the appropriate test for two classifiers evaluated on the same items (Dietterich, 1998), and it is far more sensitive than comparing marginal accuracies. We state the sensitivity explicitly, since it is quoted throughout. Writing d for the discordant-pair rate between two systems, the difference detectable at $\alpha = 0.05$ and 80% power is $2.80\sqrt{d/n}$. The rate varies with how similar the two systems are. Our seed-varied models are the most similar pair we

Instrument	n	Primary use
Official test	526	Table 2, fusion
Held-out split	2,030	ablations, McNemar tests
Leave-one-out	8,156	Table 3, Fig. 1

Table 1: The three evaluation instruments used in this paper. Accuracies from different instruments are not comparable. Leave-one-out figures are the lowest of the three because the full gallery is dense with singleton classes, which is why whitening is worth 36.9 points on the official test but 18.9 points under leave-one-out.

System	top-1	top-5	top-10
Frozen HWGAT, raw cosine	0.2510	0.3441	0.3821
+ PCA whitening	0.6198	0.6958	0.7262
+ ArcFace fine-tune	0.6711	0.7205	0.7357
+ TCN score fusion	**0.6825**	0.7243	0.7357

Table 2: Official test leaderboard progression. Our final system ranked 3rd on the official leaderboard by top-1 accuracy, and 2nd by top-10 accuracy. The system ranked 2nd by top-1 reached 0.7319 on top-10.

compare, agreeing on 96.1% of held-out queries, which gives $d \approx 0.039$ and a minimum detectable effect of 1.23 points at $n = 2{,}030$. Comparisons between genuinely different systems have higher discordance. Across the pairs in this paper d reaches roughly 0.12, which raises the threshold to 2.15 points. We quote 1.23 as the best case available to us rather than a figure that applies uniformly, and the corresponding floors are 2.41 points on the 526-clip test split and 0.61 points under leave-one-out.

The instrument tracked the leaderboard closely. It gave 62.81% against an actual 61.98%, and 66.70% against an actual 67.11%. This gives us reasonable confidence that a null on the held-out split is also a null on the test set, rather than an artefact of a different distribution.

Table 1 names the three instruments and states which results depend on each. We flag one contamination case for honesty. A model trained on all 8,156 clips has seen every held-out query. Held-out numbers for that model are inflated and are not comparable to the rest of the table. We judge that model on the leaderboard alone.

6 Results

Table 2 gives the progression on the official test split. Two interventions produced gains the official test split can resolve, and a third, score fusion, was adopted for our submission but falls inside that split's noise band, as Section 9 sets out. Fifteen others produced no usable gain. Twelve were indistinguishable from zero and three were harmful, as Section 10 sets out.

One feature of our final standing is worth drawing out. Our top-1 of 0.6825 placed 3rd, but our top-10 of 0.7357 placed 2nd, ahead of the system that beat us on top-1 by 1.7 points. The spread between our top-1 and top-10 is 5.3 points. For the winning system it is 2.1. Our retrieval surfaces the correct class within ten candidates about as reliably as the best submission, and loses ground specifically in ordering the top of that list. This is what one would expect if the residual limitation is fine-grained ranking among near neighbours rather than gross representational failure. It is consistent with the whitening analysis in Section 7. Correcting global variance restores coarse retrieval, but the one-shot prototypes remain noisy enough that the top of the ranking stays unstable.

The three interventions we adopted share a property. Each changed the representation. Whitening changed its geometry, fine-tuning changed the encoder, and the TCN introduced a second architecture with different failure modes. Every inference-side method that reused an existing representation returned zero, regardless of sophistication. That set includes CSLS, query expansion, reranking, test-time augmentation, and alternative inference rules. We return to this pattern in Section 11.

7 Variance Equalisation

Whitening the frozen cross-corpus encoder moved test top-1 from 0.2510 to 0.6198. This section asks what the transform is actually correcting, and how its contribution relates to the contribution of fine-tuning.

7.1 Rescaling versus axis removal

The obvious explanation is that whitening removes a handful of nuisance directions. Signer identity, framing and lighting all plausibly dominate the leading principal components of a keypoint encoder. That explanation is wrong here, and the whitening grid says so (Figure 1). Sweeping the whitening strength α at fixed d moves leave-one-out top-1 by 29.0 points. Sweeping d at the operating point of $\alpha = 1.0$ moves it by 0.4 points, with $d = 2$, $d = 4$ and $d = 8$ falling within 0.08 points of each other.

The one regime where discarding components does matter is $\alpha = 0$, a pure rotation with no rescal-

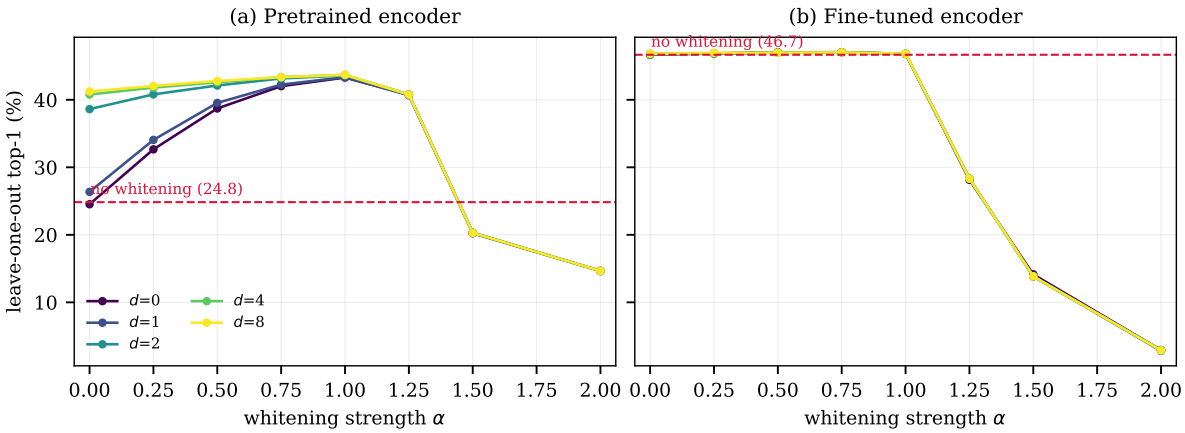


Figure 1: Leave-one-out top-1 against whitening strength α, with one line per number of discarded leading components d. Dashed line is the unwhitened baseline. On the pretrained encoder (a) the curves rise steeply with α and converge, so rescaling rather than deletion carries the gain, and the exception at $\alpha = 0$ is where deletion is the only correction available. On the fine-tuned encoder (b) whitening is flat until it becomes harmful, which is the substitution effect of Section 7.2 seen directly. Both encoders collapse under over-whitening.

ing, where moving from $d = 0$ to $d = 8$ is worth 16.7 points. This is the exception that confirms the rule. When no rescaling is applied, deleting the dominant axes outright is the only available way to stop them from controlling the inner product. Once α is large enough to rescale them, deleting them adds almost nothing.

The pathology is therefore global scale imbalance across the whole space, rather than a few contaminated axes. This mirrors the anisotropy reported for contextual word embeddings (Ethayarajh, 2019), where a small number of dominant directions likewise distort cosine geometry and can be corrected by removing or rescaling them (Mu and Viswanath, 2018). Cosine similarity implicitly assumes comparable variance across dimensions. When a cross-corpus encoder has a steep variance spectrum, a small number of directions dominate every inner product. The remaining dimensions carry the lexical distinctions, and they are effectively ignored. Equalising variance restores their contribution.

Over-whitening is harmful, and sharply so. Accuracy peaks at $\alpha = 1.0$ on the pretrained encoder and then falls away, reaching 20.3 points at $\alpha = 1.5$ and 14.7 at $\alpha = 2.0$, well below the unwhitened baseline of 24.8. Pushing genuinely low-variance noise axes to unit scale amplifies them. The same collapse appears on the fine-tuned encoder, and at a lower threshold. Reducing dimensionality to 256 or 128 also costs 2 to 3 points, which confirms that the discriminative signal is distributed rather than concentrated.

Encoder	no whitening	whitening	gain
Pretrained	24.84	43.69	+18.85
Fine-tuned	46.67	47.05	+0.38

Table 3: Leave-one-out top-1 (%) over the 8,156-clip gallery. Whitening is worth +18.85 points on the pretrained encoder and +0.38 on the fine-tuned one, an interaction of −18.5. The fine-tuned row is inflated by training contamination and should not be read as a clean estimate of the effect of fine-tuning.

7.2 Substitution with fine-tuning

If whitening and fine-tuning correct the same anisotropy, they should not compose. We tested this with a 2×2 over pretrained and fine-tuned encoders crossed with whitening, evaluated throughout by leave-one-out retrieval over the 8,156-clip gallery so that every cell comes from one instrument (Table 3).

Whitening is worth +18.85 points on the pretrained encoder and +0.38 points on the fine-tuned one. The interaction is −18.5. Practically the whole value of the transform disappears once the encoder has been fine-tuned. Panel (b) of Figure 1 shows the same result as a nearly flat line. Across every setting of d and every α up to 1.0, whitening moves the fine-tuned encoder by well under a point, so the shallow optimum reported below sits inside a band that is immaterial in practice.

We are deliberately careful about what this table does and does not license. The two within-row

comparisons are clean, because each contrasts an encoder with itself. The across-row comparison is not, because the fine-tuned model was trained on these gallery clips and has seen every query. Its row is inflated by an unknown amount. We therefore make no claim that either intervention has the larger main effect. Our claim is about their relationship, which the within-row contrasts establish on their own: the marginal value of whitening collapses by 98% once fine-tuning has been applied.

We should be explicit about the alternative reading this leaves open. A model that has memorised the gallery depends less on cosine geometry for retrieval, and would therefore show a flat whitening response whether or not its embedding space is still anisotropic. The flat fine-tuned row is thus consistent with substitution and with saturation by memorisation, and this table cannot separate the two. The decisive experiment is a fine-tuned encoder evaluated on queries it has never seen, on the official test split or on a held-out split withheld from training. We regard that as the first thing to run in any extension of this work.

The point can be put more strongly. On the instrument that produced it, the $+0.38$ residual sits below the 0.61-point detection floor derived in Section 5. Whitening the fine-tuned encoder does not merely help less. It produces no difference this evaluation can resolve. The contrast with the pretrained row is stark, since $+18.85$ clears the same floor by a factor of thirty.

The direction of the asymmetry is what makes this practically useful. Both interventions correct the same pathology, but they cost very different amounts. The whitening transform is a single SVD fitted in seconds on cached embeddings. The fine-tune is roughly an hour of GPU time in our setup and considerably more on larger backbones. When two interventions substitute for each other, the rational order is to try the cheap one first and measure before paying for the expensive one.

A dose-response pattern in the optimal whitening strength corroborates the substitution account, though for reasons given below it is not independent evidence for it. The pretrained encoder prefers aggressive correction at $d = 4$ and $\alpha = 1.0$. The fine-tuned encoder peaks milder, at $d = 0$ and $\alpha = 0.75$, and tolerates over-whitening far worse. A model trained on all 8,156 clips prefers milder correction again. The optimum moves toward weaker correction as the encoder acquires in-domain data, which is what a better-conditioned embedding space would produce. We note that the middle point of this gradient is not resolvable on this instrument, since the fine-tuned encoder's curve is shallow enough over $\alpha \leq 1.0$ that the location of its peak sits inside the same band as the residual itself. The gradient therefore rests on its endpoints.

We state the limit of this argument rather than leaving it to be found. All three optima are measured by leave-one-out over the gallery, and across these three encoders the amount of in-domain training data and the degree of contamination with respect to that evaluation increase together. An encoder that has memorised more of the gallery depends less on cosine geometry and would also prefer a milder correction, so the gradient on this instrument is consistent with the substitution account and with contamination alike. Separating them would require sweeping α on the official test split, whose queries no encoder trained on. We did not run that sweep. We therefore present the gradient as consistent with the substitution account rather than as independent support for it.

8 Input Masking and Arm Geometry

Sign phonology decomposes signs into handshape, location, movement and orientation. We asked which of these the backbone actually relies on. To do so we restricted the input to hand keypoints alone and fine-tuned under that mask.

HWGAT collapsed, losing approximately 20 points as recorded in Table 5. A TCN trained from scratch on identical masked inputs was essentially unaffected.

One confound must be stated before this comparison can be read. HWGAT was pretrained on full-body skeletons and then fine-tuned under the mask, whereas the TCN was trained from scratch under it. A model trained from scratch shapes its filters around whatever inputs exist, while a pretrained model arrives with an attention structure organised around groupings it no longer receives. The contrast may therefore be measuring pretrained-then-deprived against trained-under-deprivation rather than architectural dependence as such. Separating the two would require training HWGAT from scratch under the mask, which we did not have the compute budget to add.

Subject to that caveat, we read the result as a statement about the architecture rather than about ISL. The windowed graph attention structure is

built around groupings of head, arm and hand. The arm keypoints are not merely informative features. They are the scaffolding that the attention mechanism is organised around. Remove them and the inductive bias is not degraded but invalidated. The TCN treats keypoints as undifferentiated channels and simply routes around the missing input.

This bears on transfer to other sign languages. An architecture whose performance is contingent on a specific articulator grouping will transfer unpredictably to corpora with different framing conventions, different use of signing space, or seated rather than standing capture. We encourage reporting masked-input results as a routine robustness check in sign language recognition papers. The check is cheap and it is diagnostic.

9 Score Fusion

We attempted seven fusions and exactly one improved the score (Table 4). Whether that improvement is real is the subject of the closing paragraph of this section. The predictor of success was not architecture family, not individual member accuracy, and not fusion weight. It was pairwise top-1 agreement between members.

Members agreeing on 88 to 92% of queries contributed nothing or hurt. They fail on the same clips, so averaging their scores mostly averages their noise. The TCN agreed on 60 to 64% of queries and was worth $+1.14$ points.

Diversity alone is not sufficient either. Transformers agreeing on only 48 to 52% added nothing. Those members were both diverse and weaker. Score fusion cannot extract signal from a member that is wrong in novel ways without also being right in novel ways.

The useful summary is that ensemble headroom saturated after one architecturally distinct member. Fusion weight was insensitive across the range 0.6 to 0.9, which is itself evidence that the fused signal was not delicately balanced.

One consequence of Section 5 should be stated plainly rather than left for a reader to derive. The $+1.14$ point fusion gain was measured on the official test split, whose detection floor we put at 2.41 points. The fusion we adopted for our final submission therefore sits inside the noise band of the instrument that selected it. We report it because it is what we submitted, not because we can demonstrate that it helped. This is an uncomfortable illustration of the argument we make about the 526-clip split, and it applies to our own decisions as readily as to anyone else's. The 1.7-point gap between our top-1 and the second-placed system falls inside the same band.

10 Negative Results

Table 5 lists them. Five deserve comment.

10.1 The largest observed differences

An ArcFace head trained on frozen features gained $+0.69$ points and proxy classification lost 0.54 points. These are the largest differences we observed among the null interventions, and both are smaller than the 1.23-point minimum detectable effect derived in Section 5, so neither is distinguishable from zero on this instrument. We report them separately for transparency rather than because they were detected. We note that the two quantities are not interchangeable. A minimum detectable effect states what a design would catch four times in five at a given power. Whether a particular observed difference reaches significance depends on the discordant-pair counts, and under McNemar a difference of this size can fall either side of the threshold. We therefore report both differences without asserting either outcome, and release the paired per-query outcomes so that the test can be recomputed. Neither changed a decision we made and we shipped neither.

10.2 Post-hoc similarity methods

CSLS (Conneau et al., 2018), alpha query expansion (Radenović et al., 2019) and rank fusion (Cormack et al., 2009) all reweight or reproject the same similarity structure. Once whitening had corrected the geometry, there was nothing left for them to correct. Their usual gains in the retrieval literature may be partly gains against uncorrected anisotropy. If so, they are redundant with whitening rather than additive to it.

10.3 Flip test-time augmentation

Mirrored queries scored 67.09% against 67.04% for originals, a difference of 0.05 points. The encoder is essentially handedness-invariant, so no residual variance remains for test-time augmentation to average away. This follows directly from training with flip probability 0.5.

The result also verifies the window-aware permutation described earlier. An incorrect permutation would produce anatomically invalid poses and a collapsed score, not parity.

Fusion	agreement	Instrument	Δ top-1
HWGAT seed variants (3)	88–92%	test	−0.19
HWGAT + full-data HWGAT	not measured	test	−0.19
Pretrained + fine-tuned	not measured	held-out	0.00
+ from-scratch TCN	60–64%	test	**+1.14**
+ 3 TCNs, 2 transformers	48–52%	test	0.00
+ restricted-view spec.	not measured	test	0.00

Table 4: Fusion outcomes, ordered by when they were attempted. Only the first architecturally distinct member improved the score, and that improvement sits below the test split's detection floor (Section 9).

Intervention	Instrument	Δ top-1
CSLS hubness correction (k=5,10,20)	LOO	+0.02
Alpha query expansion (k=2,3,5)	LOO	$\leq$ +0.02
Whitening re-sweep, fine-tuned feats	LOO	0.00
Extended fine-tuning (12→15+ ep)	held-out	+0.15
Two-encoder fusion	held-out	0.00
Flip TTA	held-out	+0.05
Retrain on all 8,156 clips	test	$\approx$ 0.00
Seed-varied HWGAT ensemble	test	−0.19
Nine-model ensemble vs. four	test	0.00
Fusion weight 0.6 → 0.9	test	0.00
Largest observed differences, still below threshold		
ArcFace head on frozen features	held-out	+0.69
Proxy classification vs. retrieval	held-out	−0.54
Harmful		
Hand-only HWGAT fine-tune	held-out	$\approx$ −20
Dimensionality reduction (256, 128)	LOO	−2 to −3
Over-whitening (α 1.25–2.0)	LOO	−2 to −3

Table 5: Interventions producing no usable gain, with the instrument each was measured on. Every difference in the first block is smaller in magnitude than the minimum detectable effect on the instrument that measured it, so none is distinguishable from zero. Those effects are 1.23 points on the held-out split, roughly 2.4 points on the official test split, and 0.61 points under leave-one-out over the 8,156-clip gallery. The two entries in the second block are the largest differences observed and remain below threshold. Failing to detect an effect is not evidence that no effect exists.

10.4 Train and gallery allocation

Training on all 8,156 clips rather than the 6,126-clip split produced no leaderboard change. The 2,030 held-out clips contributed about as much sitting in the gallery as they would have as training examples. For one-shot retrieval at this scale, the allocation of a clip between training and gallery is close to immaterial.

10.5 Coverage

Leave-one-out over the enriched gallery, bucketed by class depth, gave 33.2% top-1 for classes with one remaining example and 38.8% for classes with five or more. The ceiling at high depth is low and the relation is non-monotonic, with classes holding two examples scoring below classes holding one. That inversion is counterintuitive, since more examples per class should help, and we attribute it to the confound set out next rather than to any property of the retrieval rule.

We read this as consistent with the representation rather than data quantity being the binding constraint, and we stop short of asserting it. Class depth is not randomly assigned. How many clips a word appears in reflects how common, how short and how canonical its sign is, so difficulty is confounded with depth and a shallow gradient cannot isolate representation quality on its own. The non-monotonic dip is most likely a symptom of the same confound, since the classes that happen to hold exactly two clips are not a random sample of the vocabulary. Establishing the causal claim would require holding class difficulty fixed while varying depth, which the available annotations do not support.

11 Discussion

A pattern runs through the null results. Every intervention that reused an existing representation at inference time returned nothing, and every intervention that changed the representation returned something. The largest difference among the representation-reusing interventions was $+0.69$ points from training a new head on frozen features, which is below the detection threshold on the instrument that measured it. This suggests a triage rule for one-shot retrieval under compute constraints. Before spending time on inference-side sophistication, establish whether the representation has slack in it. Whitening is a cheap probe for exactly that.

The 526-clip test set cannot distinguish systems separated by less than roughly 2.4 points, a floor that encompasses most of the gaps on the final leaderboard and our own fusion gain. Shared tasks in low-resource sign language processing face a real tension here, because test data is expensive to collect and annotate. We nonetheless encourage organisers to report a noise floor alongside the leaderboard, and participants to build a held-out instrument rather than iterating against a split that cannot support it. We suspect that a meaningful fraction of reported improvements in this space fall within noise, a concern raised more generally for NLP evaluation by Card et al. (2020).

Our findings concern one backbone, one corpus and one sign language, and we have not tested the whitening result beyond that setting. The mechanism it rests on, anisotropy in a cross-corpus embedding space, is not ISL-specific and is well documented elsewhere, so we expect the effect to appear wherever a pretrained encoder is applied across corpora. We state that as an expectation rather than a demonstrated result. The magnitude will depend on how anisotropic the source encoder happens to be. The masking result is explicitly architecture-specific, and we make no claim beyond HWGAT.

12 Conclusion

We reached 3rd place on the WSLP 2026 ISLR shared task with a system whose largest single component is a closed-form linear transform. Whitening a frozen cross-corpus encoder and fine-tuning it end-to-end turn out to be substitutes rather than complements. Each corrects the same embedding anisotropy, and applying either one collapses the marginal value of the other, with an interaction of -18.5 points. Fifteen further interventions, several of them standard practice in retrieval, produced no usable gain. Twelve were indistinguishable from zero at 1.23-point detection sensitivity and three were harmful. For researchers working on under-represented sign languages with limited compute, the practical takeaway is that two interventions with very different price tags can buy the same thing, so the cheap one is worth measuring before the expensive one is paid for.

Limitations

Our results come from a single sign language, a single pretrained backbone and a single shared task. We cannot separate properties of HWGAT from properties of ISL keypoint data without replication on other encoders and corpora.

The held-out split is drawn from the same gallery distribution as the training data and shares signers with it. It may therefore understate the difficulty of genuinely held-out signers. The close agreement with leaderboard scores is reassuring but not decisive.

Our 2×2 is evaluated by leave-one-out retrieval over the gallery, and the fine-tuned model was trained on those clips. That row is inflated by contamination, so the table supports our claim about the relationship between whitening and fine-tuning, which rests on within-row contrasts, but not any claim about which intervention has the larger main effect. Separating the two main effects cleanly would require training a fine-tuned encoder that has never seen the evaluation clips at all, which we did not have the compute budget to add. We also note that leave-one-out accuracy over the full gallery is systematically lower than official test accuracy, because the gallery contains many singleton classes, so numbers from the two instruments should not be compared directly.

Our negative results are powered to detect effects above 1.23 points on the held-out split and 0.61 points under leave-one-out. We cannot rule out smaller true effects, several of which might in principle compose.

Gallery enrichment used a second corpus whose recording conditions differ from the task data. We did not isolate how much of the enrichment gain comes from coverage and how much from domain diversity.

Finally, we report single-run results for the fine-tuning experiments. We did not have the compute budget for multi-seed confidence intervals on every

configuration. The seed-variance experiments we did run were aimed at ensembling rather than at characterising run-to-run spread.

Artifact and Code Availability

All corpora used are publicly released research resources, obtained from their original distributors and used within their stated terms. The backbone architecture and its pretrained weights are publicly available from the authors of the original work. Our contribution consists of keypoint extraction and caching scripts, the whitening fit and sweep, the retrieval and fusion code, the held-out evaluation harness including the McNemar procedure, and the configuration files needed to reproduce every number reported here. These will be released under an open licence on acceptance, and are available from the authors on request in the meantime.

Ethics Statement

Sign language recognition systems built without the involvement of Deaf communities risk encoding hearing assumptions about how signing works. They also risk producing tools that serve researchers more than signers. Our work is a technical contribution to a shared task and makes no claim to deployment readiness. Top-1 accuracy near 0.68 on isolated signs is far from adequate for any communicative application, and presenting it as such would be harmful.

References

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7784–7793.

Dallas Card, Peter Henderson, Urvashi Khandelwal, Robin Jia, Kyle Mahowald, and Dan Jurafsky. 2020. With little power comes great responsibility. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9263–9274. Association for Computational Linguistics.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *International Conference on Learning Representations (ICLR)*.

Gordon V. Cormack, Charles L. A. Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms Condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 758–759.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. 2019. ArcFace: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4690–4699.

Thomas G. Dietterich. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pages 55–65. Association for Computational Linguistics.

Hervé Jégou and Ondřej Chum. 2012. Negative evidences and co-occurrences in image retrieval: The benefit of PCA and whitening. In *European Conference on Computer Vision (ECCV)*, pages 774–787.

Abhinav Joshi, Ashwani Bhat, Pradeep S, Priya Gole, Shashwat Gupta, Shreyansh Agarwal, and Ashutosh Modi. 2022. CISLR: Corpus for Indian Sign Language recognition. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10357–10366, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Abhinav Joshi, Romit Mohanty, Mounika Kanakanti, Andesha Mangla, Sudeep Choudhary, Monali Barbate, and Ashutosh Modi. 2024. iSign: A benchmark for Indian Sign Language processing. In *Findings of the Association for Computational Linguistics: ACL 2024*. Association for Computational Linguistics.

Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. 2020. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1459–1469.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. MediaPipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

Jiaqi Mu and Pramod Viswanath. 2018. All-but-the-top: Simple and effective postprocessing for word representations. In *International Conference on Learning Representations (ICLR)*.

Suvajit Patra, Arkadip Maitra, Megha Tiwari, K. Kumaran, Swathy Prabhu, Swami Punyeshwarananda, and Soumitra Samanta. 2024. Hierarchical windowed graph attention network and a large scale dataset for isolated Indian sign language recognition. *arXiv preprint arXiv:2407.14224*.

Filip Radenović, Giorgos Tolias, and Ondřej Chum. 2019. Fine-tuning CNN image retrieval with no human annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(7):1655–1668.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Advaith Sridhar, Rohith Gandhi Ganesan, Pratyush Kumar, and Mitesh Khapra. 2020. INCLUDE: A large scale dataset for Indian sign language recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1366–1375.

Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.

Tom Vandendriessche, Mathieu De Coster, Annelies Lejon, and Joni Dambre. 2025. Representing signs as signs: One-shot ISLR to facilitate functional sign language technologies. *arXiv preprint arXiv:2502.20171*.

WSLP 2026 Shared Task Organisers. 2026. Findings of the WSLP 2026 shared task on isolated sign language recognition. In *Proceedings of the 2nd Workshop on Sign Language Processing (WSLP 2026)*. To appear.

What Actually Transfers? Diagnosing Output-Projection Initialization in Low-Resource Sign Language Translation

Kai-Yang Peng **Chunki Lim** **Shu-Kai Hsieh**
Graduate Institute of Linguistics, National Taiwan University
{r14142015, r14142001, shukaihsieh}@ntu.edu.tw

Abstract

Released sign-to-text checkpoints combine visual and text-side parameters, so intact-checkpoint comparisons cannot identify what drives transfer. We study this confound in low-resource Taiwan Sign Language (TSL) by separating and recombining the visual and mT5 components of Uni-Sign checkpoints trained on Chinese Sign Language (CSL)-to-Chinese and American Sign Language (ASL)-to-English data, then fine-tuning the combinations identically on TSL. The CSL-derived visual initialization improves content-word accuracy by 15.0 points with its released mT5 parameters, but changes accuracy by only -0.5 points with untouched mT5-base, showing that the measured benefit of visual pretraining depends on the text-side initialization. We assess video dependence by replacing target clips with duration-matched held-out clips at evaluation. Further interventions localize much of the accuracy and video-dependence effects to the untied output projection: replacing an ASL-to-English model's projection with mT5-base's improves accuracy by 25.0 points and increases sensitivity to clip replacement. Permutation and timing controls show that the effect depends on pretrained row-to-token correspondence and on the projection being present during fine-tuning. Absolute translation quality remains low, so we interpret these experiments diagnostically. The results show that released checkpoints should be analyzed as structured initializations whose components can interact, rather than treated as indivisible evidence of cross-language transfer.

1 Introduction

Transfer studies in low-resource sign language translation (SLT) often rely on available pretrained checkpoints as starting points. A released checkpoint, however, combines parameters from both the signed-video and text-generation components, together with the optimization history that jointly shaped them. If the source sign language also covaries with the written output language across checkpoints, comparisons between intact models cannot separate these sources. This problem arises in multilingual and cross-language SLT (Asasi et al., 2026; Yin et al., 2022; Tanzer and Zhang, 2025) and in systems that combine separately pretrained visual and text-generation components (Zhang et al., 2023; Zhou et al., 2023; Wong et al., 2024).

Taiwan Sign Language (TSL) provides a particularly clear setting for examining this confound. The available TSL corpus pairs signed utterances with Chinese translations, while TSL is historically linked to Japanese Sign Language rather than Chinese Sign Language (CSL) (Smith, 2005; Fischer and Gong, 2010; Su and Tai, 2009; Sasaki, 2007). Uni-Sign (Li et al., 2025) releases checkpoints with the same architecture for CSL-to-Chinese and ASL-to-English, but not for TSL. Adapting one of these checkpoints therefore changes the provenance of the visual and text-side components together. An apparent transfer advantage could reflect the visual initialization, the text-side initialization, or their interaction.

We separate these possibilities by decomposing each checkpoint into the parts shown in Figure 1. We call the pose encoder and its linear bridge to mT5 the *visual initialization*. The *text-side initialization* comprises the remaining mT5 parameters, including the *output projection* that maps final hidden states to vocabulary logits. In standard mT5 this projection is tied to the input embedding, but the released Uni-Sign checkpoints store and train it as a separate `lm_head` matrix. We label each initialization by its source checkpoint. Thus, the "CSL-Daily text-side initialization" denotes the mT5 portion of the CSL-Daily CSL-to-Chinese checkpoint (Zhou et al., 2021); "mT5-base" denotes the published multilingual model before sign-language training. In projection swaps, the *recipient* supplies the rest of the model and the *donor*

Proceedings of the Second Workshop on Sign Language Processing, pages 43–53
October 29, 2026 ©2026 Association for Computational Linguistics

supplies only the substituted output projection.

Our intervention strategy follows work on identifying transferable components (Neyshabur et al., 2020), model stitching and parameter editing (Bansal et al., 2021; Ilharco et al., 2023), output-embedding specialization (Chung et al., 2021), and cross-lingual lexical reinitialization (Artetxe et al., 2020; de Vries and Nissim, 2021; Minixhofer et al., 2022; Dobler and de Melo, 2023). Few-sample fine-tuning motivates explicit checks for optimization sensitivity (Zhang et al., 2021), and signed-language NLP motivates cautious evaluation (Yin et al., 2021; Müller et al., 2023). Unlike WECHSEL or FOCUS, we keep the vocabulary, tokenizer, architecture, and TSL adaptation task fixed while varying the parameters used for initialization.

We proceed from coarse to fine interventions. First, we test whether a fixed visual initialization contributes equally under different text-side initializations. We then isolate the output projection and use random, rescaled, and row-permuted controls to identify which of its properties matter. Finally, we vary replacement timing and learning rate to distinguish an initialization effect from a better final readout or a fixed checkpoint ranking. Accordingly, our causal interpretation is limited to output-projection initialization under low-resource adaptation. The experiments do not isolate Chinese versus English output-language pretraining as the causal factor.

2 Setup

Data and splits. We use the Taiwan Sign Language Corpus (Ministry of Culture, Taiwan, 2025), which contains 407 filmed paragraphs, 5,282 annotated utterances with Chinese translations, and 25 signers. We split at the paragraph level because adjacent utterances often share a signer, topic, and material from the same clause. Each of three folds uses 243–245 paragraphs (3,176–3,255 utterances) for training and 135–137 (1,760–1,761) for testing. Appendix G gives the per-fold counts. The split is not signer-disjoint: at most 1 test signer per fold is absent from training. The reported results therefore measure generalization to held-out *paragraphs*, not held-out *signers*.

Forced-choice diagnostic. BLEU is near the floor for every model in this study, so we use a forced-choice diagnostic that remains informative when free translation quality is low. It contains 940 four-way items spanning 238 content

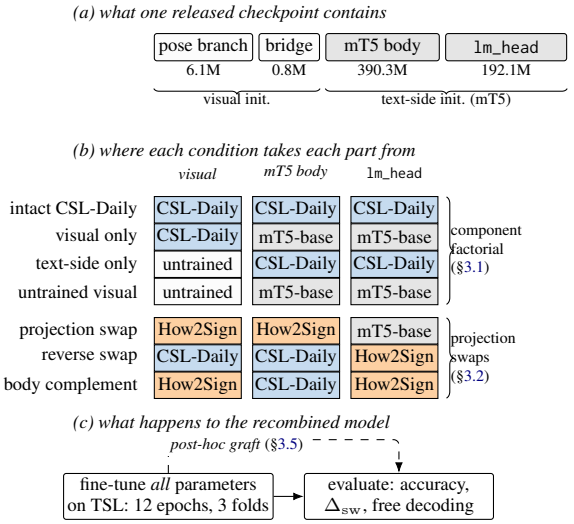


Figure 1: Intervention protocol. (a) A released UniSign checkpoint is partitioned into a pose branch, the bridge into mT5, the mT5 body, and the untied output projection (`lm_head`). (b) Each row specifies the source checkpoint for each component. The first four rows show the core 2×2 component factorial; §3.1 adds the shared stage-1 visual initialization as a provenance control. The final three isolate the output projection and its complement in §3.2. (c) All recombined models are fine-tuned on TSL and evaluated under the same procedure. The dashed path marks the post-hoc replacement in §3.5, which installs the projection only after fine-tuning.

words. Each item replaces one content-word slot in a reference sentence with the gold candidate or one of three distractors. Distractors match the gold in corpus part of speech, are close in log frequency, and avoid exact location-and-handshape collisions (Appendix A). For a candidate c tokenized as $y_1, \ldots, y_T$ and a video v, we score each token conditional on the candidate's own preceding tokens and define

$$s(c \mid v) = \frac{1}{T} \sum_{t=1}^{T} \log p_\theta(y_t \mid y_{<t},\, v)\,. \qquad (1)$$

The model selects $\arg\max_c\, s(c \mid v)$. Conditioning on the candidate prefix avoids dependence on previous model predictions; mean log probability controls for candidate length. Absolute accuracy is not by itself evidence of visual grounding. One rank-balanced Qwen2.5-1.5B-Instruct mean-token baseline is exactly 25% (Qwen Team, 2024), while other text-only scorers given the sentence without video reach 38.0–55.3%. We therefore use accuracy as a task-level measure and base claims about video dependence on paired input interventions.

Dependence on clip identity. We measure whether predictions depend on the target video by replacing that video with a different, duration-matched clip from another held-out paragraph while leaving the sentence and candidate set unchanged. Let A_{target} and A_{swap} denote accuracy with the original and substituted clips. We define *wrong-clip sensitivity* as $\Delta_{\text{sw}} = A_{\text{target}} - A_{\text{swap}}$. Positive values therefore indicate that predictions benefit from the identity of the target clip. We prefer a real substitute to an all-zero pose sequence because the latter lies outside the training distribution and can even reverse the measured effect (Appendix A). A near-zero Δ_{sw} should not be interpreted as evidence that no visual information is represented internally. It can also arise when a decoder has collapsed to nearly constant output. We therefore interpret Δ_{sw} together with free decoding and the bidirectional parameter interventions. A second 975-item diagnostic targeting referential pointing signs provides an independent check (Appendix A).

In tables, Acc. denotes A_{target}, while A_{swap} is the mean accuracy over three duration-matched substitute clips. Accuracy contrasts in the text subtract A_{target} between model cells; factorial interactions subtract those contrasts. Contrasts use paired unrounded scores, so subtraction of displayed one-decimal cells can occasionally differ by 0.1 point from the reported contrast.

Models, training, and statistics. In Uni-Sign, the visual initialization comprises a 6.1M-parameter pose branch over COCO-WholeBody keypoints (Jin et al., 2020; Jiang et al., 2023) plus a 0.8M-parameter bridge into mT5-base (Xue et al., 2021). We use released CSL-Daily, CSL-News, How2Sign, OpenASL, and WLASL checkpoints (Zhou et al., 2021; Li et al., 2025; Duarte et al., 2021; Shi et al., 2022; Li et al., 2020). All inherit stage-1 CSL-News pose pretraining, so the "CSL-Daily visual initialization" includes that shared stage plus later CSL-Daily adaptation (§3.1). The untied $250{,}112 \times 768$ output projection contains 192.1M parameters and is stored separately from the input embedding, allowing independent replacement. RAND-VIS, the no-visual-pretraining baseline, pairs an untrained pose branch with untouched mT5-base.

Unless noted otherwise, conditions are fine-tuned end to end for 12 epochs with Uni-Sign's published learning rates over three paragraph folds; key projection replacements are rerun with three seeds on fold 0. Confidence intervals use paragraph-clustered bootstrap resampling and paired contrasts use paragraph-level permutation tests. Full hyperparameters and checkpoint provenance are reported in Appendix G.

	CSL-Daily		mT5-base	
visual init.	Acc.	Δ_{sw}	Acc.	Δ_{sw}
CSL-Daily (stage 1 + adapt.)	56.5	+22.1	39.6	-0.5
CSL-News (stage 1 only)	56.0	+22.2	40.2	+1.3
untrained	41.5	+1.6	40.1	+0.0

Table 1: Factorial comparison, pooled over three folds. Acc. is target-clip accuracy; Δ_{sw} is the percentage-point drop under duration-matched real-clip substitution. The middle row isolates shared stage-1 CSL-News pose pretraining from later CSL-Daily adaptation.

3 Results

We first establish whether visual and text-side initializations interact, then localize that dependence and test what makes the output projection useful. We finally test generality, replacement timing, free generation, and optimization sensitivity.

3.1 Visual pretraining depends on the text-side initialization

Table 1 reports cell-level Acc. and Δ_{sw}. The contrasts here use Acc., not Δ_{sw}. With the CSL-Daily text-side initialization, the visual-pretraining contrast is $56.5 - 41.5 = +15.0$ points ([+12.0, +17.9]); with mT5-base it is $39.6 - 40.1 = -0.5$ points ([-2.2, +1.2]). Their difference is the interaction: $+15.0 - (-0.5) = +15.5$ points ([+12.1, +18.9], $p < 0.001$). The corresponding Δ_{sw} values are $+22.1$ versus $+1.6$ in the first column and -0.5 versus $+0.0$ in the second, showing the same conditional pattern in clip dependence.

The middle row locates the accuracy interaction earlier in Uni-Sign training. Stage-1 pose pretraining contributes $56.0 - 41.5 = +14.5$ points with the CSL-Daily text-side initialization, while later CSL-Daily adaptation adds only $56.5 - 56.0 = +0.5$ points. With mT5-base, stage-1 pretraining contributes $40.2 - 40.1 = +0.1$ points. The conditional benefit is therefore already present after shared pose pretraining rather than being specific to CSL-Daily visual adaptation.

Cross-loading the two Chinese-output checkpoints leaves both combinations near their intact scores (56.0% and 56.0%). Thus the interaction

does not require the original jointly trained pairing, motivating a finer intervention on the text-side initialization.

3.2 The interaction localizes to the output projection

We use parameter displacement only to choose the first text-side component to intervene on. Relative to mT5-base, released input embeddings differ by only 0.0015–0.0025 in relative L_2, whereas the released output projections differ by 0.92–0.98. These distances are descriptive and are not treated as causal explanations.

Replacing only How2Sign's output projection with the mT5-base projection raises content-word accuracy by 25.0 points ([+21.2, +28.9]) and increases wrong-clip sensitivity by $+14.7$ points ($[+11.7, +17.7]$, $p < 0.001$). The reverse intervention has the opposite effect: replacing CSL-Daily's projection with How2Sign's lowers accuracy from 56.5% to 24.3% and reduces Δ_{sw} from $+22.1$ to $+0.3$ points. Its residual wrong-clip effect differs from RAND-VIS by only $+0.3$ points ($[-1.0, +1.6]$, $p = 0.668$); the interval includes zero, so these data do not show a detectable separation between the two conditions. Both directions replicate across three seeds.

A complement control asks whether the surrounding mT5 parameters can reproduce the same effect without changing the projection. How2Sign receives all CSL-Daily mT5 parameters except the output projection while retaining its own projection. This complement contains 390.3M distinct parameters, about twice the 192.1M parameters in the projection, but reaches only 25.5% accuracy with $\Delta_{\text{sw}} = +2.1$ and remains degenerate under free decoding. The much larger surrounding component therefore does not reproduce the rescue. We consequently localize much of the text-side dependence to the output projection, without treating it as the only parameter that can matter.

3.3 Token-aligned rows are necessary for a useful projection

Localization to the output projection still leaves several explanations open. The successful donors differ from the original How2Sign projection in whether they are pretrained, their overall scale, their script-specific row statistics, and the correspondence between vocabulary rows and the tokens scored by those rows. Resetting the projection gives a modest $+3.4$-point gain over How2Sign's original projection ([+0.4, +6.3]), so removing that initialization can help. The norm-matched random projection nevertheless remains 21.6 points below mT5-base ([-25.7, -17.5]). Changing the random initialization's scale changes accuracy by only $+0.4$ points ([-1.6, +2.4]), while rescaling How2Sign's own projection changes accuracy by at most about one point (Appendix B).

Recipient	output-projection donor	Acc.	Δ_{sw}
How2Sign	own	23.9	+0.5
	random, norm-matched	27.3	+2.0
	mT5-base, rows permuted	27.4	+2.0
	CSL-Daily	48.5	+14.3
	mT5-base	48.9	+15.2
OpenASL	own	24.6	+0.7
	CSL-Daily	49.7	+17.3
	mT5-base	55.7	+22.6
WLASL	own	26.4	+1.8
	CSL-Daily	45.9	+10.8
	mT5-base	50.0	+15.6
CSL-Daily	own	56.5	+22.1
	How2Sign	24.3	+0.3
	mT5-base	63.0	+29.0

Table 2: Output-projection replacements at initialization. The recipient supplies all parameters except the output projection, and the named donor supplies only that matrix. All other parameters and the training procedure are held fixed. Columns report content-word accuracy and wrong-clip sensitivity in percentage points; every row pools the same three folds. Contrasts quoted in the text use paired, unrounded scores.

The row-permutation control directly tests token correspondence. Before fine-tuning, we randomly permute the vocabulary rows of the mT5-base projection. This preserves the exact multiset of rows, its Frobenius norm, row-norm distribution, singular values, and conditioning, while changing which token each pretrained row scores. The input embedding is left unchanged. If the recipient needed only generic pretrained matrix structure, the permuted donor should perform like the intact donor.

Instead, the permuted projection reaches 27.4%, compared with 48.9% for the same matrix without permutation. The 21.5-point loss is large ([-25.5, -17.5]), and wrong-clip sensitivity falls to $+2.0$ points. Relative to the norm-matched random projection, the estimated difference is only $+0.1$ points ([-2.7, +2.9]); relative to How2Sign's original projection it is modestly positive ([+0.8, +6.3]). Permuting rows only within script groups also fails to recover the intact donor (24.8%, [-29.7, -16.7]). The useful property is therefore not global pre-

trained structure or script-level scale. The rows must remain aligned with the tokens they were pretrained to score.

This does not identify lexical semantics as the mechanism, because row-to-token correspondence remains confounded with other token-specific mT5 geometry. Consistent with that narrower interpretation, a gloss-to-Chinese projection trained without video also improves How2Sign, and improvement does not track video provenance or distance from mT5-base.

3.4 The effect replicates across recipients

The same three-fold replacement pattern appears on OpenASL and WLASL (Table 2). The mT5-base projection improves content-word accuracy by 31.2 and 23.6 points, respectively ([+27.4, +35.0]; [+19.3, +28.0]); CSL-Daily's projection yields gains of 25.1 and 19.5 points, with wrong-clip sensitivity rising from near zero in both. The effect is therefore not How2Sign-specific, although our evidence remains within the Uni-Sign architecture.

3.5 The projection must be present during adaptation

A timing intervention tests whether the donor is merely a better final readout. Installing CSL-Daily's projection before fine-tuning improves How2Sign by 24.6 points, whereas replacing the projection in the fully trained model changes accuracy by only -1.5 points. Across three folds, the post-hoc graft leaves wrong-clip sensitivity near zero (+0.3 points, $[-0.9, +1.4]$, $p = 0.673$).

A 768×768 ridge map fitted on held-out development states provides a stronger readout control. It lowers donor cross-entropy below How2Sign's own projection and recovers 11.8 accuracy points, yet Δ_{sw} remains near zero. Thus a better final linear readout can improve lexical scoring without reproducing the training-time change in clip dependence. The main effect requires the donor to be present during adaptation (Appendix C).

3.6 Generation and optimization provide complementary evidence

Unconstrained decoding shows the same ordering. How2Sign with its original projection obtains BLEU 0.01 and only 2.4% distinct outputs, indicating near-constant generation. With mT5-base it reaches BLEU 2.07 and 77.3% distinct outputs. A norm-matched random projection reaches BLEU 0.11 with 11.6% distinct outputs, an intermediate stock-fragment regime (Appendix F). CSL-Daily improves from BLEU 1.93 to 4.38 with mT5-base, whereas the How2Sign projection yields BLEU 0.00 with a 93.9% modal output. The replacement can therefore prevent generation collapse without producing high-quality TSL translation.

Equation 1 can remain informative when autoregressive decoding collapses because it scores each candidate given its own prefix. A learning-rate sweep provides a separate optimization check: at 3×10^{-4}, intact How2Sign differs from RAND-VIS by -1.2 points ([-7.6, +4.8]), while the projection effect remains 25.2 points ([+19.8, +30.7]). Appendix D reports the full sweep.

4 Discussion and conclusion

A fixed visual initialization is strongly beneficial with the CSL-Daily text-side initialization and nearly neutral with mT5-base, showing why intact checkpoints can obscure component interactions. The stage-1 condition further shows that this conditional benefit is already present after the shared CSL-News pose-pretraining stage, before CSL-Daily-specific visual adaptation. Because that stage itself uses CSL-News, this locates the interaction without establishing language-general visual pretraining. Much of the dependence localizes to the output projection: one-matrix replacements improve three ASL-derived recipients, the reverse replacement damages CSL-Daily, and the twice-larger complement does not reproduce the effect. Row permutation and timing show that token-aligned pretrained rows must be present during adaptation; learning-rate sensitivity shows that initialization is not the whole account.

More broadly, when checkpoint components have different training histories, the checkpoint name is too coarse a unit of analysis. Transferability is better treated as a property of a component in its initialization context than in isolation. This separates what a component carries into adaptation from whether the surrounding initialization lets it become useful. Future checkpoint comparisons should therefore report component provenance and, where feasible, include recombination controls before attributing differences to source-language transfer. These methodological claims do not establish language matching, semantic grounding, or whether generation collapse causes low wrong-clip sensitivity.

Limitations

The evidence comes from one TSL corpus, one Uni-Sign/mT5 architecture, and one low-resource adaptation setting. Checkpoint provenance and written output language are not independently varied across the released checkpoints, so the experiments do not identify a Chinese-versus-English language effect. The untied 192.1M-parameter output projection is also specific to this architecture.

Our splits are constructed at the paragraph level rather than by signer. At most 1 test signer per fold is absent from training, so the evaluation measures generalization to held-out paragraphs and not to unseen signers. Whether the same effects hold under a genuine cross-signer split remains untested and would be necessary for stronger claims about deployment.

The row-permutation experiment establishes the importance of row-to-token correspondence but does not identify which token-specific property is causal. Lexical-semantic organization and other geometry learned during mT5 pretraining remain entangled within that correspondence. The experiment therefore narrows the mechanism without fully specifying it.

The forced-choice diagnostic also retains substantial text-side signal. Matched text-only scorers reach 38.0–55.3%, and a scorer with additional discourse context reaches 66.2%. Moreover, generation collapse can itself reduce measured wrong-clip sensitivity, so the diagnostic does not fully separate these phenomena. BLEU remains low even after generation recovers. Some auxiliary donor controls—including text-only and within-script permutation variants—as well as learning-rate cells and mechanistic probes use fold 0 only; the main random and full-permutation controls now pool all three folds. A fold-0 label-smoothing check does not rescue intact How2Sign (Appendix D), but other hyperparameters are not systematically ablated.

Ethical Considerations

The Taiwan Sign Language Corpus (Ministry of Culture, Taiwan, 2025) is a research-use beta release. We redistribute neither corpus material nor signer metadata, and all reported split statistics are aggregate counts. Pose keypoints can retain identity-related cues, so we handle them as potentially identifying data. The location-and-handshape filter used in the diagnostic is an automated corpus check rather than linguistic validation. It cannot replace review by fluent TSL signers or participation by Deaf communities (Yin et al., 2021; Müller et al., 2023). The diagnostic design and analysis did not include consultation with Deaf signers, and we therefore make no claims about usability or community benefit.

References

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Sobhan Asasi, Ozge Mercanoglu Sincan, and Richard Bowden. 2026. SIGNET: Motion-level knowledge transfer for cross-language sign language translation. In *European Conference on Computer Vision (ECCV)*. To appear. arXiv:2606.28626.

Yamini Bansal, Preetum Nakkiran, and Boaz Barak. 2021. Revisiting model stitching to compare neural representations. In *Advances in Neural Information Processing Systems*, volume 34, pages 225–236.

Hyung Won Chung, Thibault Févry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. 2021. Rethinking embedding coupling in pre-trained language models. In *International Conference on Learning Representations (ICLR)*.

Wietse de Vries and Malvina Nissim. 2021. As good as new. How to successfully recycle English GPT-2 to make models for other languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 836–846, Online. Association for Computational Linguistics.

Konstantin Dobler and Gerard de Melo. 2023. FOCUS: Effective embedding initialization for monolingual specialization of multilingual models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13440–13454, Singapore. Association for Computational Linguistics.

Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giró-i-Nieto. 2021. How2Sign: A large-scale multimodal dataset for continuous American Sign Language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2735–2744.

Susan Fischer and Qunhu Gong. 2010. Variation in East Asian sign language structures. In Diane Brentari, editor, *Sign Languages*, pages 499–518. Cambridge University Press, Cambridge.

GLM Team. 2024. ChatGLM: A family of large language models from GLM-130B to GLM-4. *Preprint*, arXiv:2406.12793.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. In *International Conference on Learning Representations (ICLR)*.

Tao Jiang, Peng Lu, Li Zhang, Ningsheng Ma, Rui Han, Chengqi Lyu, Yining Li, and Kai Chen. 2023. RTMPose: Real-time multi-person pose estimation based on MMPose. *arXiv preprint arXiv:2303.07399*.

Sheng Jin, Lumin Xu, Jin Xu, Can Wang, Wentao Liu, Chen Qian, Wanli Ouyang, and Ping Luo. 2020. Whole-body human pose estimation in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 196–214.

Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. 2020. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1459–1469.

Zecheng Li, Wengang Zhou, Weichao Zhao, Kepeng Wu, Hezhen Hu, and Houqiang Li. 2025. Uni-Sign: Toward unified sign language understanding at scale. In *International Conference on Learning Representations (ICLR)*.

Ministry of Culture, Taiwan. 2025. Taiwan Sign Language Corpus. Beta release, research use. Accessed 2026-08-05. `https://tslcorpus.moc.gov.tw/`.

Benjamin Minixhofer, Fabian Paischer, and Navid Rekabsaz. 2022. WECHSEL: Effective initialization of subword embeddings for cross-lingual transfer of monolingual language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3992–4006, Seattle, United States. Association for Computational Linguistics.

Mathias Müller, Zifan Jiang, Amit Moryossef, Annette Rios, and Sarah Ebling. 2023. Considerations for meaningful sign language machine translation based on glosses. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 682–693, Toronto, Canada. Association for Computational Linguistics.

Behnam Neyshabur, Hanie Sedghi, and Chiyuan Zhang. 2020. What is being transferred in transfer learning? In *Advances in Neural Information Processing Systems*, volume 33, pages 512–523.

Qwen Team. 2024. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Daisuke Sasaki. 2007. Comparing the lexicons of Japanese Sign Language and Taiwan Sign Language: A preliminary study focusing on the difference in the handshape parameter. In David Quinto-Pozos, editor, *Sign Languages in Contact*, pages 123–150. Gallaudet University Press, Washington, DC.

Bowen Shi, Diane Brentari, Gregory Shakhnarovich, and Karen Livescu. 2022. Open-domain sign language translation learned from online video. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6365–6379, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Wayne H. Smith. 2005. Taiwan Sign Language research: An historical overview. *Language and Linguistics*, 6(2):187–215.

Shiou-fen Su and James H.-Y. Tai. 2009. Lexical comparison of signs from Taiwan, Chinese, Japanese, and American Sign Languages: Taking iconicity into account. In James H.-Y. Tai and Jane Tsay, editors, *Taiwan Sign Language and Beyond*, pages 149–176. The Taiwan Institute for the Humanities, National Chung Cheng University, Chia-Yi, Taiwan.

Garrett Tanzer and Biao Zhang. 2025. YouTube-SL-25: A large-scale, open-domain multilingual sign language parallel corpus. In *International Conference on Learning Representations (ICLR)*.

Ryan Wong, Necati Cihan Camgoz, and Richard Bowden. 2024. Sign2GPT: Leveraging large language models for gloss-free sign language translation. In *International Conference on Learning Representations (ICLR)*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Aoxiong Yin, Zhou Zhao, Weike Jin, Meng Zhang, Xingshan Zeng, and Xiaofei He. 2022. MLSLT: Towards multilingual sign language translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5109–5119.

Kayo Yin, Amit Moryossef, Julie Hochgesang, Yoav Goldberg, and Malihe Alikhani. 2021. Including signed languages in natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7347–7360, Online. Association for Computational Linguistics.

Biao Zhang, Mathias Müller, and Rico Sennrich. 2023. SLTUNET: A simple unified model for sign language

translation. In *International Conference on Learning Representations (ICLR)*.

Tianyi Zhang, Felix Wu, Arzoo Katiyar, Kilian Q. Weinberger, and Yoav Artzi. 2021. Revisiting few-sample BERT fine-tuning. In *International Conference on Learning Representations (ICLR)*.

Benjia Zhou, Zhigang Chen, Albert Clapés, Jun Wan, Yanyan Liang, Sergio Escalera, Zhen Lei, and Du Zhang. 2023. Gloss-free sign language translation: Improving from visual-language pretraining. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20871–20881.

Hao Zhou, Wengang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. 2021. Improving sign language translation with monolingual data by sign back-translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1316–1325.

A Diagnostics and statistical procedure

The content-word diagnostic replaces one reference slot with the gold word or one of three distractors. Gold and distractor tokens share the corpus POS tag, and distractor types are selected near the gold in log frequency. Candidates must be standalone tokens under jieba and must not share the gold sign's location and handshape signature. That filter removes exact sign-phonological collisions, but near-minimal pairs remain: an audit finds 16.6% of distractors share the gold location, 6.3% share its handshape, and 47.0% of items contain at least one such distractor. These are automated corpus checks, not human judgments.

The gold occupies each Qwen rank equally often, which makes the construction-specific Qwen mean-log-probability baseline exactly 25%. That guarantee does not generalize to other scorers. Under the same sentence-level input, Qwen with summed log probability reaches 55.3%, a GLM-family scorer (GLM Team, 2024) reaches 38.0%, and a unigram prior reaches 38.3%. With preceding sentences from the same paragraph Qwen reaches 66.2%, but the video conditions receive no such discourse context, so this is an unmatched reference.

For the wrong-clip intervention, the substituted clip comes from another held-out paragraph and is matched in duration to the target. The median duration difference is 9 ms, and results average over three substitute clips. A within-paragraph substitution, which holds signer and session fixed, still reduces CSL-Daily accuracy by 18.9 points ([+15.9, +21.8]). The effect therefore persists when signer, session, and paragraph context are held more closely matched. We do not use an all-zero clip as the primary intervention because it lies outside the training distribution and can even reverse the direction of the measured effect.

The pointing-sign diagnostic. The second diagnostic contains 975 four-way items targeting referential pointing signs. It is constructed by the same procedure, with the varied slot placed at a referential expression rather than a content word. The items are more predictable from text alone, so absolute accuracy is less informative than on the content-word set. Their wrong-clip effects are substantially larger, however, which makes the set a useful independent check.

Every main-text intervention produces the same direction of effect on this set. Intact How2Sign scores 45.7% with $\Delta_{\text{sw}} = -1.2$. Installing mT5-base's projection raises accuracy to 85.8% and Δ_{sw} to +40.0, compared with +15.2 on the content-word set. In the reverse direction, intact CSL-Daily scores 77.6% with $\Delta_{\text{sw}} = +33.7$. Under How2Sign's projection it falls to 46.7% and $\Delta_{\text{sw}} = +0.0$, eliminating measured clip dependence. Frame shuffling, which removes temporal order while preserving the frame distribution, reduces intact CSL-Daily accuracy by 24.8 points and the reverse-replacement condition by 0.3.

The two item sets therefore agree on the direction and the localization of every intervention we report, differing mainly in how much of the available wrong-clip effect each resolves. §D notes the one place they disagree, which concerns additivity with the learning rate rather than the projection effect itself.

For each cell, Acc. is the proportion of diagnostic items whose gold candidate ranks first. Swap accuracy is computed for each of three duration-matched substitute clips and averaged; Δ_{sw} subtracts this mean from target-clip accuracy. Accuracy contrasts subtract paired cell accuracies on shared items, and interaction contrasts subtract two such paired differences. All main content-word confidence intervals use 10,000 bootstrap samples resampled over paragraphs; paired p-values use paragraph-level sign-flip permutation tests. Thus the paragraph, not the item, is the resampling unit. A pooled three-fold content-word contrast contains 940 items from 341 paragraphs, and the paragraph count determines the effective sample size. Holm correction is applied to the original family

of reported contrasts. The component interaction remains significant after correction, whereas the small encoder-only content-word contrast does not.

B Full donor and reset controls

How2Sign `lm_head` donor	Acc.	Δ_{sw}
own	23.9	+0.5
mT5-base	48.9	+15.2
mT5-base, rows permuted	27.4	+2.0
mT5-base, permuted in script	24.8	+2.0
CSL-Daily	48.5	+14.3
CSL-News	46.4	+14.3
TSL text	49.1	+14.5
gloss→Chinese	47.3	+10.7
TSL text, drift-matched	43.0	+11.4
random, norm-matched	27.3	+2.0

Table 3: Every output-projection donor tested on the How2Sign recipient, with content-word accuracy and wrong-clip sensitivity. The CSL-News, TSL-text, gloss-to-Chinese, drift-matched, and within-script-permutation rows are fold 0 only; the norm-matched random and full-permutation rows pool three folds. Paired contrasts quoted in the text compare cells on shared items.

All neutral or Chinese-capable pretrained donors improve How2Sign, and the estimated gains overlap broadly despite their different provenance. The gloss-to-Chinese projection improves accuracy by 22.7 points ([+16.6, +29.2]). Its estimated difference from mT5-base is -0.6 points, with an interval spanning zero ([-3.2, +2.0]). The TSL-text donor labeled "drift-matched" was trained until its projection reached a relative L_2 distance of 0.913 from mT5-base, comparable to the released fine-tunes. It still improves How2Sign, but its development cross-entropy is 2.63. Because this displacement was obtained partly by overtraining, the donor is not a clean matched instrument.

The random reset distinguishes pretraining from mere removal of the English projection: a norm-matched random projection is 21.6 points below mT5-base ([-25.7, -17.5]). Changing the random initialization's row-norm scale changes accuracy by only $+0.4$ points ([-1.6, +2.4]), so scale alone does not explain the gap to mT5-base. Rescaling How2Sign's own projection before training, using CJK-row correction, radius correction, or both, changes accuracy by at most about one point and leaves wrong-clip sensitivity near zero. The same CJK correction can move accuracy when applied post-hoc, suggesting a test-time calibration effect rather than a successful training initialization.

Appendix E reports a complementary gradient probe before any parameter update.

The permutation donors are described in §3.3. Both are built from mT5-base's own projection with the vocabulary rows reordered, so their row multiset, Frobenius norm, row-norm distribution and singular values are identical to mT5-base's by construction. The full permutation also changes the script region associated with each row, whereas the within-script permutation does not.

C Post-hoc readout tests

A post-hoc replacement re-scores a fully trained recipient without further training. Pooled over the three folds, replacing How2Sign's projection with CSL-Daily's changes accuracy by -1.5 points. Across three fold-0 seeds, the corresponding changes are -0.6, $+1.5$, and -3.6 points, with score-margin changes of -0.223, -0.209, and -0.207 nats. The self-replacement control changes little, and a post-hoc mT5-base projection also gives small mixed effects across three seeds. The same mT5-base projection at initialization improves How2Sign by 25.0 points and CSL-Daily by 6.5 points.

We additionally fit 768×768 interface maps on held-out development states, to test whether a donor projection simply expects a different hidden-state basis. How2Sign and CSL-Daily decoder states are indeed far apart. A ridge map lowers the CSL-Daily donor's dev cross-entropy on How2Sign states below the recipient's own projection and recovers about 12 points of contrastive accuracy, yet Δ_{sw} stays near zero; related orthogonal and cross-entropy-trained maps show the same dissociation. On CSL-Daily the same kind of aligned replacement leaves a reduced but nonzero wrong-clip effect. The alignment can therefore repair a lexical readout while leaving measured wrong-clip sensitivity near zero in the trained How2Sign endpoint.

D Optimization and model-selection controls

The mT5 learning-rate sweep changes the interpretation of the intact ASL baseline but not the projection effect. On fold 0, How2Sign with its own projection scores 25.2% at 10^{-4} and 39.4% at 3×10^{-4}; with mT5-base's projection the corresponding cells reach 56.7% and 64.5%. The paired projection effects are large at all three rates. At

3×10^{-4}, the projection still improves accuracy by 25.2 points.

Both diagnostics show that a higher learning rate can restore wrong-clip sensitivity, but they differ on whether the learning-rate and projection effects are additive. On content words the projection and learning-rate effects remain separable. On pointing signs the additional neutral-projection gain at the higher rate is $+5.1$ points ($[-0.4, +10.7]$, $p = 0.074$). We therefore avoid claiming a universal additive mechanism.

A fold-0 run with label smoothing set to zero leaves intact How2Sign at 23.9%, a -0.6-point change from the default setting ([-3.3, +2.0]), with $\Delta_{\text{sw}} = -0.3$. Removing label smoothing alone therefore does not rescue this baseline.

The projection effect also persists when model selection is held to the final epoch. Three cells were rerun while saving epoch 12 in addition to the best-dev checkpoint. The effect is 22.7 points at the best-dev epochs and 24.8 points at epoch 12 ([+18.0, +32.1]). Thus the gap is not removed by evaluating both runs at the same final epoch.

E Gradient probe

Before any parameter update, we backpropagate one training step through 12 shared batches for eight recipient/projection combinations, measuring the visual-component gradient share, $\|(g_{\text{pose}}, g_{\text{bridge}})\|/\|g_{\text{total}}\|$, where g_{bridge} is the pose-to-mT5 bridge gradient. The four configurations that later collapse lie between 0.390 and 0.510; the four that later avoid collapse lie between 0.555 and 0.634. The ordering has no overlap, but it should not be read as a continuous predictor: CSL-Daily with mT5-base's projection has a lower visual-component share than intact CSL-Daily while reaching higher final accuracy. The probe shows only that the training signal begins differently.

Absolute gradients clarify two distinct failures. How2Sign with a random projection has a very large pose gradient (34.14) but an even larger mT5-body gradient (82.62), so this share is low because the decoder requires repair. CSL-Daily with How2Sign's projection instead reduces the pose gradient to 2.52 from 6.78 under its own projection, a small absolute gradient on the visual components. Across-batch gradient cosine stays near zero in all conditions, so this coherence measure provides no separation between the two outcome groups in these runs.

F Free decoding and a noncollapsed control

Cell	BLEU	distinct	modal
How2Sign, own	0.01	2.4	34.2
How2Sign + mT5-base	2.07	77.3	1.7
How2Sign + permuted	0.19	25.2	8.7
How2Sign + random	0.11	11.6	15.0
How2Sign + CSL-Daily	1.36	86.4	2.0
CSL-Daily, own	1.93	85.6	3.0
CSL-Daily + mT5-base	4.38	88.6	1.3
CSL-Daily + How2Sign	0.00	1.2	93.9
How2Sign + CSL body, own proj.	0.00	0.9	95.4

Table 4: Unconstrained generation, fold 0. "Distinct" and "modal" are the percentage of output strings that are unique and the share taken by the single most frequent string. A near-zero BLEU with high distinctness and a near-zero BLEU with one repeated string are qualitatively different failure modes, which BLEU alone does not separate.

Projection	Illustrative free-decoding output
own	"我。。。。。。。。。。" (degenerate)
random	"我會去" (stock fragment)
mT5-base	"怎麼用手語溝通?" (multiword sentence)

Table 5: The three output regimes made concrete: hand-selected How2Sign fold-0 outputs illustrating qualitative differences that a near-zero BLEU hides. All quantitative claims rest on the full-set statistics above, never on these examples.

Table 5 illustrates the three generation regimes: near-constant output, repeated stock fragments, and multiword sentences after projection replacement.

The complement control takes CSL-Daily's whole mT5 half except `lm_head` and places it on How2Sign. It scores 25.5% with $\Delta_{\text{sw}} = +2.1$ on the content-word diagnostic and remains degenerate under free decoding (BLEU 0.00, distinct 0.9%). The large surrounding mT5 component therefore does not substitute for the projection on either diagnostic or on generation.

To test whether the component ordering is peculiar to collapsed video translation, we replace pose features with gold gloss sequences and train the same mT5 to Chinese for 12 epochs, varying only the initial `lm_head`. The neutral projection reaches dev CE 2.050, BLEU 26.46 and 95.8% distinct outputs. The CSL-Daily projection reaches BLEU 11.99, the How2Sign and OpenASL projections

reach 1.89 and 1.79, and the norm-matched random projection reaches 1.02 with 33.1% distinct outputs. After training, the task-specific projections move only about 0.12–0.15 in relative L_2 from their own initializations. The same broad ordering therefore persists in a regime where the top model is not degenerate, and the output projection remains strongly anchored to where it started.

G Checkpoint provenance and reproducibility

Splits. The corpus holds 407 paragraphs (207 monologue, 200 dialogue), 5,282 utterances and 25 signers. Across the three folds, training takes 243–245 paragraphs (3,176–3,255 utterances), development 27 (267–345), and test 135–137 (1,760–1,761). Folds are balanced on paragraph type and diagnostic-item count by greedy largest-first assignment, with ties broken on a hash of the paragraph identifier, so they are deterministic. Signer identity is not used as a splitting variable: at most 1 test signer per fold does not also appear in that fold's training split, so these are held-out-paragraph splits and not cross-signer ones. We report signer allocation only in aggregate, consistent with redistributing no signer metadata.

Provenance and hyperparameters. Uni-Sign initializes each downstream sign checkpoint from stage-1 pose-only pretraining on CSL-News. The released mT5 bodies stay close to this common ancestor while `lm_head` moves much further. Input embeddings shift only about 0.0015–0.0025 in relative L_2 from mT5-base, whereas released output projections shift by about 0.92–0.98. Script-wise row norms also differ between CSL and ASL checkpoints, but the random, rescaling and permutation controls in Appendix B are inconsistent with script scale as a sufficient explanation of the training-time improvement.

Unless otherwise noted, training uses AdamW, batch size 8 with two-step gradient accumulation, a pose-branch learning rate of 3×10^{-4}, an mT5 learning rate of 3×10^{-5}, 5% warmup and label smoothing 0.2. Clips are capped at 192 frames and targets at 64 tokens. The learning-rate sweep changes only the mT5 rate. Cells are selected by development cross-entropy, with the final-epoch control of Appendix D. All reported distractor candidates are unseen as diagnostic labels during training. Code, analysis scripts and the per-run measurements behind every number reported here are available at `https://github.com/lopentu/wslp-transfer-artifact`. The diagnostic items are released as metadata and construction scripts rather than as text: no corpus material or signer metadata is redistributed, and no checkpoint weights are, since the donors carry no redistribution license.

LSC-CARE: An annotated dataset for medical Catalan Sign Language

Carlos Escolano[1] **Joan Gomà Cortés**[1] **Gemma Barberà**[2]
Gerard Sant[3] **Maite Melero**[4]

[1]Universitat Politècnica de Catalunya [2]Universitat Pompeu Fabra
[3]University of Zurich [4]Barcelona Supercomputing Center

carlos.escolano@upc.edu, joan.goma.cortes@estudiantat.upc.edu,
gemma.barbera@upf.edu, gerard.santmuniesa@uzh.ch, maite.melero@bsc.es

Abstract

Deaf signers face persistent communication barriers in healthcare settings, where reliance on third-party interpreters can compromise privacy precisely when confidentiality matters most. We present LSC-CARE, the first annotated dataset of Catalan Sign Language (LSC) for the medical domain, comprising 2,784 signed segments (approximately 4.5 hours) produced by 11 deaf native signers across ten emergency-care topics. Each segment is paired with time-aligned ID-gloss annotations produced by a deaf native linguist and written Catalan translations produced by an LSC interpreter, following conventions consistent with the LSC SignBank. To assess the dataset's utility for Sign Language Translation, we benchmark three settings: a gloss-free end-to-end video-to-text model, a cascade of sign language recognition and gloss-to-text translation, and an oracle gloss-to-text upper bound. The cascade consistently outperforms the end-to-end approach despite a high gloss error rate, and, unlike it, is statistically invariant to signer identity, while the 21-BLEU gap between cascade and oracle identifies visual sign recognition as the main bottleneck for low-resource medical SLT. We release the dataset to support accessible healthcare technologies for the Catalan Deaf community.

1 Introduction

Catalan Sign Language (*llengua de signes catalana*, LSC) is a low-resource language. It is used in Catalonia, Andorra and the island of Menorca by 25,000 signers in their daily life. Of these, about 12,000 are deaf individuals who use it as their primary language of communication, while the remaining users are hearing individuals, including family members and professionals, such as interpreters, teachers and linguists.[1]. In the field of machine Sign Language Translation (SLT), dominant paradigms based on large volumes of data are not directly applicable to LSC due to a "double minoritization" (Quer, 2012). Despite the existence of LSC linguistic resources (Barberà et al., 2018) and the open-access publication of the first reference corpus of Catalan Sign Language, the Corpus LSC (Institut d'Estudis Catalans, 2025), we still face a shortage of data annotated with the necessary conventions to be interpretable by Artificial Intelligence (AI) systems. Research highlights the complexity of generating photorealistic and linguistically appropriate signed sequences (Bragg et al., 2019); (Rastgoo et al., 2021). To address this gap, approaches are identified that use "core glosses" as an intermediate representation to shift complexity toward the inferential capacity of Large Language Models (LLMs) and preserve translation coherence (Müller et al., 2023).

This paper presents LSC-CARE, the first annotated corpus of Catalan Sign Language for the medical domain, together with the first translation baselines trained on it. Our contributions are threefold. First, we release a corpus of 2,784 signed question–answer segments across ten emergency-care topics, produced by 11 deaf native signers and annotated with time-aligned ID-glosses and written Catalan translations. Second, we benchmark three translation settings — gloss-free end-to-end, gloss-based cascade, and an oracle gloss-to-text upper bound — establishing reference results for medical LSC translation. Third, we provide a statistical analysis of length, topic, and signer effects that characterises where each architecture fails, showing that intermediate gloss supervision yields both higher translation quality and robustness to signer identity under extreme data scarcity. The full dataset can be downloaded from: https://huggingface.co/

[1]Data sourced from the Catalan Federation for the Deaf (FESOCA), officially cited by the Catalan Secretariat for Language Policy (Generalitat de Catalunya) in https://tuit.cat/8HN3H

Proceedings of the Second Workshop on Sign Language Processing, pages 54–69
October 29, 2026 ©2026 Association for Computational Linguistics

datasets/projecte-aina/LSC-CARE

2 Related Work

Sign language corpora. Progress in SLT has been driven by a small number of annotated benchmarks. RWTH-PHOENIX-Weather 2014T (Camgöz et al., 2018; Forster et al., 2014) pairs roughly 11 hours of German Sign Language weather broadcasts with glosses and German text; CSL-Daily (Huang et al., 2018) provides about 23 hours of Chinese Sign Language across everyday topics; How2Sign (Duarte et al., 2021) offers around 80 hours of instructional American Sign Language. At larger scale, YouTube-SL-25 (Tanzer and Zhang, 2025) mines over 3,000 hours of weakly aligned captioned video across 25+ sign languages, trading annotation quality for volume. A common trait of the annotated corpora is domain restriction: each covers one narrow communicative context, and models trained on them transfer poorly beyond it. For LSC specifically, available resources include the reference Corpus LSC (Barberà et al., 2015; Institut d'Estudis Catalans, 2025), but no resource targets the medical domain, and none provides the gloss-level annotation density required for cascade SLT. LSC-CARE fills this gap at the extreme low-resource end of the landscape: at 4.5 hours it is less than half the size of PHOENIX14T, while covering a domain (emergency healthcare) where translation errors carry direct consequences for patient safety.

Translation architectures. Computational approaches divide into gloss-based and gloss-free paradigms. Gloss-based cascades decompose the task into continuous sign language recognition, typically trained with CTC (Graves et al., 2006), followed by gloss-to-text translation (Camgöz et al., 2018); TwoStream-SLR/SLT (Chen et al., 2022) represents the state of the art, fusing RGB and pose-keypoint streams before CTC decoding. The paradigm benefits from modular supervision and diagnosable errors, but inherits an information ceiling: glosses systematically discard non-manual markers, spatial grammar, and prosody (Müller et al., 2023). Gloss-free end-to-end systems (Zhou et al., 2023) avoid the annotation bottleneck and have become dominant where large data exists (Tanzer and Zhang, 2025; Zhang et al., 2024), but jointly learning visual parsing, alignment, and generation demands training volumes far beyond what low-resource languages can offer. Which paradigm degrades more gracefully under extreme scarcity remains an open empirical question.

2.1 LSC-CARE: Data Compilation

After doing research into the emergency protocols of the public hospitals in Catalonia, we compiled a set of possible doctor–patient interactions in emergency settings. We recreated possible question-answer pairs between doctor and patient and recorded the LSC version of each sentence. Signer EB is the dedicated question signer, while the remaining 10 signers provide different possible answers to each question. The topics selected were the following: personal data; triage; general; traumatology; gynecology; pediatrics; ophthalmology; otorhinolaryngology; dermatology; pneumology. Table 1 shows the distribution of LSC-CARE by anonymised signer and medical domain.

The recruited deaf signers watched the recorded LSC questions and proposed two or three possible answers to that question. The dataset was therefore augmented in terms of linguistic expressions, lexicon and dialectal variation. This increased the dataset's linguistic and lexical diversity, as well as its dialectal variation across gender, age, and region. The dataset was professionally recorded in an audiovisual recording studio at the university sign language lab at a resolution of 747x420 or 560x420 pixels and 25 fps (see Figure 1). The studio has a chroma key background to clearly distinguish the signer. Only one signer per shot was recorded.

2.2 Deaf native signers profiles

A total of 11 deaf signers were recorded. They are all native signers, born in deaf families and attended bilingual or specific schools, depending on the age. They are all affiliated to a deaf club and therefore their usual interaction language is LSC. Including native signers allows to describe natural grammatical structures at their optimal stage of neurocognitive development, establishing a reliable baseline against which to compare other developmental variants or instances of second-language acquisition. This confirms that the sample of LSC included in the LSC-CARE contains morphosyntactic structures complex enough to be considered a baseline. Moreover, a balanced amount of gender variation and age groups was achieved, belonging to the three age groups established in the LSC Corpus (Barberà et al., 2015), namely the first group age from 18 to 30 years old, the second age group from 31 to 50 years old and the third age group from 51 to 80 years old. Table 2 summarises the

Signer ID	Dat.	Tri.	Tra.	Gen.	Gyn.	Ped.	Opht.	Oto.	Der.	Pne.	Total
EB(Q)	8	7	37	17	8	12	11	12	21	13	146
BF	-	-	74	-	-	-	-	-	72	-	146
CW	72	-	-	39	-	-	-	41	-	-	152
DC	-	-	-	-	-	-	-	76	71	-	147
DT	19	39	99	46	34	48	54	67	59	30	495
FB	44	22	131	65	23	36	30	37	96	57	541
FW	-	68	-	-	-	70	-	-	-	-	138
GW	-	54	-	-	-	-	-	-	-	75	129
NU	65	45	-	-	51	-	-	-	-	100	261
TT	65	-	244	-	-	-	-	-	109	73	491
YB	43	-	75	-	-	-	-	-	-	20	138
Total	316	235	660	167	116	166	95	233	428	368	2784

Table 1: Number of segments per anonymised signer and medical domain in LSC-CARE. The row *Total* reports the column sum per domain; the column *Total* reports the row sum per signer. [*] The abbreviations for each column are as follows: Dat. = personal data; Tri. = triage; Tra. = traumatology; Gen. = general; Gyn. = gynecology; Ped. = pediatrics; Opht= ophthalmology; Oto. = otorhinolaryngology; Der. = dermatology; Pne. = pneumology.

	Male	Female	Total
18-30 y.o.	BF	CW	2
31-50 y.o.	FB; TT	FW; DC	4
51-80 y.o.	YB; GW	EB; DT; NU	5
Total	5	6	11

Table 2: Signer's profiles according to age and gender.

different profiles.

2.3 Data characteristics and transcription

The dataset is distributed in Parquet format, with each instance representing a segmented video recording mapped to both its linguistic annotations and target natural language translation. Every sample includes precise video tracking via the signal field—which specifies the local file path to the corresponding .mov or .mp4 video—alongside signal_start and signal_end timestamps in seconds to define the exact temporal boundaries of the segment. For linguistic supervision, the glosses field provides human-annotated sign glosses transcribed in sequential order, while the output field contains the finalized clinical sentence translation in written Catalan.

The glosses used in the annotation process of the LSC-CARE belong to the list of tokens of the LSC database project developed at the Institut d'Estudis Catalans (Quer, 2017) and are part of the project under development LSC SignBank. Using the list of tokens ensures that the glosses used in the annotation remain consistent with the tags. Besides manually annotating each sign with the corresponding gloss and time slot, the global temporal span of the signed sentence was also defined with the corresponding written Catalan translation. Following usual technical and methodological motivations (Fenlon and Hochgesang, 2022), the ID-glosses annotation of the LSC-CARE was done by a deaf native linguist, while the corresponding Catalan translation was done by an LSC interpreter. This guarantees a correct identification of each sign with an ID-gloss and avoids possible spoken language bias of the transcription of the structure of the signed sentence.

3 Dataset Statistics

The complete dataset contains 2,784 segments totalling approximately 4.5 hours of active signing. Data were partitioned 80/10/10 into training (2,227 segments, 3.5 h), validation (278), and test (279) splits by random shuffling at the segment level with a fixed seed. The splits are therefore signer-dependent: all signers may appear in all splits. This choice is deliberate, given current SLT performance levels on corpora of this size, holding out entire signers would leave too little training signal, but it means reported results reflect seen-signer conditions, a point we return to in the Limitations (Section 9).

The training split contains 26,145 gloss tokens over 3,502 types. The distribution is heavily Zipfian (estimated slope -1.11): the most frequent gloss, the first-person index IX1, alone accounts

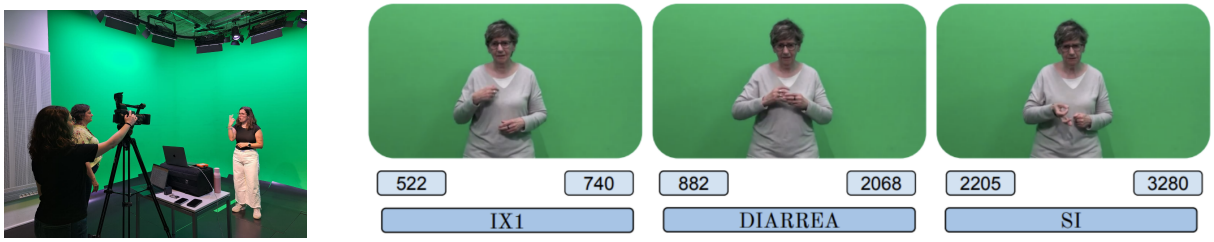


Figure 1: Examples of how the dataset was recorded. On the left, the recording setup. On the right, an example from the dataset for the sentence *Yes, I do have diarrhea.*

for 7.0% of all tokens, the 100 most frequent types cover 49.1% of the corpus, while 43.1% of the vocabulary (1,511 types) are singletons and 77.5% of types occur fewer than five times. Mean gloss sequence length is 11.8 glosses per segment (median 10, range 1–66), and mean gloss articulation duration is 0.56 s (10th–90th percentile 0.15–1.07 s).

The annotation convention encodes morphology directly in the label. Plain lexical glosses (e.g. *FEBRE, PASTILLA*) comprise 29.7% of types but 54.5% of tokens; hyphenated compound and modified forms — directional indices such as IX1-BD, classifier constructions such as CL-B, lexicalised compounds such as *TOCAR-COR* — comprise 68.1% of types, of which 83.1% occur fewer than five times, forming most of the long tail. A small set of fingerspelled sequences (75 types, 0.6% of tokens), covering names, medication names, and contact details, is retained as single gloss units following the original annotation. This distribution, a small dense head and a vast sparse tail dominated by productive morphology, mimics an actual scenario where patients may employ infrequent or even out-of-vocabulary expressions.

4 Methodology

We have measured the usefulness of our dataset on the task of Sign Language Translation (SLT) on the following different settings (for detailed hyperparameters, consult section A):

End-to-End: A direct Video-to-Text neural model that translates sign language video directly into target text without an explicit gloss supervision bottleneck.

Cascade: A two-stage pipeline consisting of a Video-to-Gloss recognition model followed by a Gloss-to-Text translation model.

Oracle: A baseline model utilizing ground-truth (reference) glosses as input to the Gloss-to-Text translation model, serving as the empirical upper bound for the cascade architecture.

4.1 End-to-End Architecture

The End-to-End (E2E) pipeline maps an input sequence of raw video frames directly to a target written Catalan sentence without utilizing an intermediate gloss representation (Sant et al., 2025b). Built on top of the `MultimodalHugs` framework (Sant et al., 2025a) using its `video2text` module, the model connects a visual feature extractor, a learned multimodal mapper, and a sequence-to-sequence translation backbone.

4.1.1 Visual Feature Extraction

Input sign language videos are temporally subsampled, and each frame is resized to 224×224 pixels and normalized using ImageNet (Deng et al., 2009) statistics. Per-frame spatial representations are extracted using a pretrained CLIP ViT-B/32 visual encoder (Radford et al., 2021). The extracted features are the final Transformer encoder layer's contrastive projection.

4.1.2 Multimodal Mapping

To bridge the gap between frame-level visual embeddings and the token embedding space expected by the downstream translation decoder, a learned multimodal mapper projects each 512-dimensional visual embedding $x_t \in \mathbb{R}^{d_v}$ at frame t into a mapped feature vector $z_t \in \mathbb{R}^{d_m}$. We employ a linear mapper that projects the visual feature extractor continuous representations to the embedding space of the translation backbone. Different options were explored, but the following simple configuration produced the best results while being the most parameter-wise efficient:

$$z_t = W x_t + b \tag{1}$$

where $W \in \mathbb{R}^{d_m \times d_v}$ and $b \in \mathbb{R}^{d_m}$.

4.1.3 Translation Backbone

We utilize M2M-100 (418M parameters) (Fan et al., 2021) [2] as the translation backbone. The encoder input embeddings are replaced directly by the mapper's projected visual frame embeddings, while the decoder retains its pretrained multilingual sequence-to-sequence weights. Translation direction is explicitly prompted via special control tokens: an `__lsc__` token is prepended to the input sequence to indicate a sign language source, and the decoder is initialized with `__ca__` to force generation in written Catalan.

4.2 Cascade Architecture

The Cascade pipeline decomposes sign language translation into two modular stages: Sign Language Recognition (SLR / Video-to-Gloss) followed by Gloss-to-Text (G2T) translation (Camgöz et al., 2018).

4.2.1 Stage 1: Sign Language Recognition (Video-to-Gloss)

The SLR model extracts per-frame visual features using CLIP ViT-B/32. Because per-frame CLIP embeddings are temporally independent, temporal context is introduced through two stacked sequence-modeling components:

Temporal Convolutional Network (TCN): A two-layer 1D convolutional network processes the feature sequence along the time axis. The first layer subsamples the temporal dimension while projecting features into a higher dimension; the second layer applies a dilated convolution to extend the effective receptive field without further downsampling.

Temporal Transformer Encoder: The downsampled sequence is processed by a stack of Transformer encoder layers (Vaswani et al., 2017). Standard sinusoidal positional encodings are added before the first layer to inject temporal order:

$$\begin{aligned} \mathrm{PE}_{(t,2i)} &= \sin\left(\frac{t}{10000^{2i/d}}\right), \\ \mathrm{PE}_{(t,2i+1)} &= \cos\left(\frac{t}{10000^{2i/d}}\right) \end{aligned} \tag{2}$$

where d is the model dimension and t is the frame position. Pre-layer normalization is applied before each sub-layer for training stability.

A linear projection layer maps the output of the Temporal Transformer encoder to a frame-level distribution over the target gloss vocabulary (plus the CTC blank token ϵ). The recognition stage is trained alignment-free using Connectionist Temporal Classification (CTC) loss. At inference time, predicted frame alignments are collapsed into a discrete gloss sequence g via greedy or beam-search decoding.

4.2.2 Stage 2: Gloss-to-Text Translation

The predicted gloss sequence g is passed to the downstream G2T model, which consists of a fine-tuned M2M-100 (418M) backbone trained on (gloss sequence, Catalan sentence) pairs. The same control token protocol is maintained: `__lsc__` prompts the input gloss sequence and `__ca__` triggers target text generation.

4.3 Oracle Baseline

Because the Cascade pipeline decouples recognition from translation, we also evaluate the G2T model in an **Oracle** setting. In this configuration, the visual SLR model is bypassed entirely, and ground-truth (reference) glosses from the dataset are fed directly into the fine-tuned M2M-100 translation backbone. This setup isolates the Gloss-to-Text generation capacity from visual perception noise, establishing an empirical performance upper bound for the cascade paradigm.

4.4 Evaluation

Evaluation is conducted using standard lexical precision metrics (BLEU (Papineni et al., 2002), ChrF (Popović, 2015)), TER (Snover et al., 2006) and semantic/quality estimation metrics (BERTScore F1 (Zhang et al., 2020), CometKiwi (Rei et al., 2022)). Statistical significance is assessed via Pearson correlation (r), Spearman rank correlation (ρ), one-way ANOVA (St»hle and Wold, 1989), and Kruskal-Wallis non-parametric tests (Ostertagova et al., 2014).

5 Experimental Setup

All models and experimental pipelines were implemented in Python using PyTorch (Paszke et al., 2019) as the primary deep learning framework. Evaluation was performed using sacrebleu (Post, 2018) [3]. All experiments where conducted on a single NVIDIA RTX 3090 with 24GB of VRAM

[2] https://huggingface.co/facebook/m2m100_418M

[3] *"nrefs:1|case:mixed|eff:no|tok:13a|smooth:exp|version:2.5.1"*

and using 50GB of system RAM. Each training took approximately 2 days under these settings.

Due to structural differences in model architecture and infrastructure requirements, the two core translation paradigms were implemented using distinct framework ecosystems.

5.1 End-to-End Pipeline Implementation

The End-to-End (`e2e`) pipeline was built using `multimodalhugs`, an open-source extension of the Hugging Face ecosystem designed for training, evaluating, and deploying multimodal architectures within standard Hugging Face workflows. The framework provides a generalized data pipeline capable of processing heterogeneous multimodal inputs. Experiment execution is managed via YAML-based configuration files, with full integration with Weights & Biases (W&B) for real-time tracking, metric logging, checkpoint saving, and experiment resumption.

5.2 Gloss-Based Cascade Pipeline Implementation

Because existing open-source libraries provided insufficient support for the two-stage visual-to-gloss-to-text pipeline, the **Cascade** architecture was implemented from scratch using PyTorch:

Video-to-Gloss (V2G): The first-stage visual recognition network uses a custom PyTorch implementation combining the feature extraction and projection modules described in Section 4.2.

Gloss-to-Text (G2T): The second-stage text generation module directly fine-tunes a pre-trained M2M-100 (418M) model checkpoint using the native Hugging Face `Trainer` API.

Both stages of the gloss-based pipeline support metric-driven early stopping, automatic checkpointing, and W&B experiment tracking.

6 Results

In this section, we present an empirical evaluation of three translation setups: the direct End-to-End (**E2E**) model, the two-stage **Cascade** model, and the **Oracle** model (which consumes ground-truth target glosses to establish an empirical upper bound). We analyze corpus-level performance, sequence length behaviors, domain topic sensitivity, and demographic fairness across signers, age groups, and genders.

6.1 Overall System Performance and Output Lengths

Table 3 summarizes overall corpus-level performance alongside generated sentence length statistics across all three architectures. Ground-truth gloss translation (**Oracle**) outperforms all models across every metric, reaching a BLEU score of 25.97, ChrF of 48.14, BERTScore F1 of 84.28, and CometKiwi of 55.86.

The two-stage **Cascade** model consistently outperforms direct **E2E** translation on surface and lexical precision, gaining $+1.90$ BLEU points, $+4.19$ ChrF points, and $+3.68$ BERTScore F1 points. However, both architectures yield comparable overall semantic quality according to CometKiwi (42.40 for E2E vs. 41.58 for Cascade).

A notable difference emerges in generated text volume:

E2E model: generates an average prediction length of 13.22 ± 9.43 words, closely mirroring the target reference length distribution (15.09 ± 9.73 words).

Cascade model: exhibits severe sentence truncation, averaging only 9.67 ± 6.80 predicted words per sample.

Comparing **Cascade** against **Oracle** isolates the performance bottleneck introduced during visual sign recognition (Video-to-Gloss). Visual recognition errors account for large drops across all metrics: -21.17 BLEU ($25.97 \rightarrow 4.80$), -28.48 ChrF ($48.14 \rightarrow 19.66$), -11.15 BERTScore F1 ($84.28 \rightarrow 73.13$), and -14.28 CometKiwi ($55.86 \rightarrow 41.58$). Furthermore, removing video recognition errors restores prediction sequence length from 9.67 words to 14.56 ± 10.24 words in the Oracle setup.

6.2 Length Effects and Correlation Analysis

We evaluated the relationship between sequence length variables and performance metrics using sample-level Pearson (r) and Spearman (ρ) correlation tests (Table 8).

All models display statistically significant negative correlations between target lengths (Gloss Length and Reference Word Length) and n-gram precision (BLEU, $p < 0.001$). Longer input signing sequences compound translation degradation.

However, output length dynamics differ fundamentally:

Model	Avg Words	WER ↓	BLEU ↑	ChrF ↑	BS F1 ↑	CKiwi ↑
E2E	13.22 ± 9.43	–	2.90	15.47	69.45	42.40
Cascade	9.67 ± 6.80	83.24	4.80	19.66	73.13	41.58
Oracle	14.56 ± 10.24	–	**25.97**	**48.14**	**84.28**	**55.86**

Table 3: Overall Corpus Results and Prediction Length Statistics across Evaluation Pipelines (Reference Target: 15.09 ± 9.73 words, 11.04 ± 6.77 glosses).

E2E Character Overlap: In E2E, generating longer sequences strongly correlates with higher character-level precision (ChrF Pearson $r = 0.320, p < 0.001$; Spearman $\rho = 0.631, p < 0.001$). While producing longer hypotheses increases character match rates, it does not translate into higher sequence-level BLEU or BERTScore F1.

Oracle Uniform Degradation: In the Oracle system, longer generated sequences yield uniformly lower scores across all metrics ($p < 0.05$), suggesting that longer sequences remain harder to translate even when visual recognition noise is removed.

6.3 Categorical, Demographic, and Domain Analysis

To analyze system stability across topics and demographics (Signer ID, Age Group, Gender), we conducted parametric one-way ANOVA (F) and non-parametric Kruskal-Wallis (H) hypothesis tests (Table 4).

The statistical test comparisons yield three key structural insights:

Signer and Demographic Invariance in Cascade: In the **Cascade** model, performance variations across signers, age groups, and genders are statistically non-significant ($p > 0.05$). This is consistent with the intermediate gloss sequence abstracting away signer-specific visual variation, although, given the low absolute scores, part of this invariance may reflect a floor effect (see Section 9).

Demographic Sensitivity in E2E: The **E2E** architecture exhibits significant sensitivity across signers ($p < 0.001$), age groups ($p < 0.01$), and genders ($p < 0.05$). For example, the 18–30 age bracket achieves higher BLEU (2.72) and BERTScore F1 (69.24) than older signers, while female signers score higher on CometKiwi (28.07 vs. 24.41). E2E models appear to directly inherit visual domain bias without intermediate abstraction.

Origin of Demographic Disparities: In the **Oracle** system, demographic variance across Age Group and Gender drops to complete statistical non-significance ($p > 0.05$). This suggests that observed demographic disparities in sign language translation originate entirely within the visual feature extraction phase rather than downstream text translation.

7 Discussion

The comparative evaluation yields a central empirical finding: an explicit gloss bottleneck appears to be a useful inductive bias in an extreme low-resource Sign Language Translation (SLT) setting. The cascade pipeline outperforms the end-to-end model on overlap-based metrics, with comparable CometKiwi scores, despite high error rates in the initial recognition stage. This performance differential indicates that even noisy symbolic intermediates retain sufficient structural information to constrain sequence generation better than direct visual conditioning, which—when trained on approximately 3.5 hours of video—exhibits severe target-language degeneration toward fluent yet unfaithful output. Qualitatively, the primary failure mode of the end-to-end architecture is topic hallucination, characterized by syntactically fluent Catalan output detached from source content, occasionally degenerating into long templated narratives. In contrast, cascade errors are traceable directly to transcription noise within the intermediate recognition stage. Section C shows a list of examples of the most common translations errors produced by our systems.

Detailed analysis of generation errors via Translation Edit Rate (TER) alignment—which accounts for phrasal block shifts alongside standard edit distance—identifies 3,872 edit operations across 4,210 reference tokens, yielding an overall cascade TER of 91.97%. The error distribution is dominated by substitutions (43.87%, 1,847 instances) and deletions (40.02%, 1,685 instances), whereas insertions

		Cascade		End-to-End		Oracle	
Category	**Metric**	**ANOVA** F	**Kruskal** H	**ANOVA** F	**Kruskal** H	**ANOVA** F	**Kruskal** H
Topic	BLEU	1.420 (n.s.)	14.385 (n.s.)	0.421 (n.s.)	14.544 (n.s.)	1.597 (n.s.)	18.404*
	ChrF	1.813 (n.s.)	15.651 (n.s.)	0.434 (n.s.)	11.503 (n.s.)	3.001**	22.577**
	CometKiwi	1.981*	15.892*	5.444***	42.852***	3.053**	21.716**
Signer	BLEU	0.891 (n.s.)	15.283 (n.s.)	2.979**	31.874***	1.238 (n.s.)	15.154 (n.s.)
	ChrF	0.823 (n.s.)	2.907 (n.s.)	3.724***	60.757***	1.404 (n.s.)	14.714 (n.s.)
	CometKiwi	1.655 (n.s.)	13.806 (n.s.)	5.079***	47.073***	2.520**	22.073*
Age Group	BLEU	0.607 (n.s.)	2.945 (n.s.)	9.974***	11.930**	0.052 (n.s.)	0.217 (n.s.)
	CometKiwi	1.465 (n.s.)	2.303 (n.s.)	5.819**	11.868**	1.549 (n.s.)	3.361 (n.s.)
Gender	CometKiwi	3.292 (n.s.)	0.996 (n.s.)	5.887*	3.969*	0.141 (n.s.)	0.181 (n.s.)

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$, n.s. = not significant.

Table 4: ANOVA (F) and Kruskal-Wallis (H) Tests Across Domain Topics and Demographic Factors.

(4.13%, 174 instances) and block shifts (3.94%, 166 instances) occur at markedly lower frequencies. The low shift frequency indicates that word-order misalignment is minimal, indicating that phrase structure is largely preserved when target vocabulary is correctly generated. Conversely, high deletion rates reflect a sequence collapse associated with the underlying Connectionist Temporal Classification (CTC) recognition bottleneck in visually ambiguous regions. High substitution rates point to downstream language model hallucination when generating text from truncated gloss input. Performance varies across thematic domains, with higher error rates recorded in specialized categories such as Dermatology (96.67% TER) and Pneumology (94.24% TER) compared to broader categories like Triage (87.99% TER) and Personal Data (89.29% TER). Errors on fingerspelled and numeric content (e.g., dosages, contact details) remain clinically critical while contributing proportionally little to aggregate overlap metrics.

Categorical decomposition across signers reveals bounded variance (ranging from 82.67% TER for signer GW to 101.11% for CW), consistent with the hypothesis that symbolic gloss representations mitigate signer-specific visual variance. However, given the high baseline error rate, this relative invariance may partially reflect a floor effect. Furthermore, as data splits are signer-dependent, generalization to unseen signers remains unverified. Finally, oracle experiments delineate the upper bound of gloss-mediated approaches: even with error-free recognition, a subset of target sentences receives low scores, highlighting information unencoded by standard gloss sequences, such as spatial grammar, non-manual markers, and deictic references. These findings point toward two clear research directions: improving visual recognition through motion-aware encoders and pose dynamics, and enriching the intermediate representation beyond unigram glosses without removing the structured intermediate supervision required in data-constrained settings.

8 Conclusions

We presented LSC-CARE, the first annotated corpus of Catalan Sign Language for medical communication: 2,784 segments across ten emergency-care topics, signed by 11 deaf native signers and annotated with ID-glosses and Catalan translations by deaf and hearing experts. Benchmarking three translation settings, we found that a gloss-based cascade clearly outperforms a gloss-free end-to-end model under extreme data scarcity, that the cascade is robust to signer identity where the end-to-end model is not, and that the dominant bottleneck is recall in the visual recognition stage rather than gloss-to-text generation. These results establish reproducible baselines for medical LSC translation and suggest that for severely under-resourced sign languages, structured intermediate supervision buys more than its annotation cost. Future work includes motion- and pose-aware visual encoders targeting classifier constructions and dedicated letter-level fingerspelling recognition. LSC-CARE and evaluation code will be publicly released.

9 Limitations

LSC-CARE is a domain-specific resource, and its limitations follow from its design. First, the data consists of scripted question–answer interac-

tions recreated from hospital emergency protocols rather than spontaneous doctor–patient conversation; elicited signing may differ from naturalistic clinical discourse in register and turn-taking, and models trained on it may not transfer to conversational or non-medical settings.

Second, our data splits are random at the segment level, so all signers and topics are seen during training. We consider signer-independent evaluation premature at current performance levels and corpus size, but this means our results characterise seen-signer conditions only, and the cascade's signer invariance should not be read as generalisation to unseen signers.

Third, the corpus is small (4.5 hours), constraining architectural conclusions: the low absolute scores mean some null results, including aspects of the signer-invariance finding, may partly reflect floor effects. Our correlation and categorical analyses also involve many hypothesis tests without multiple-comparison correction, so individual significance values should be interpreted cautiously.

Fourth, sequences making heavy use of classifier predicates were excluded from the current annotated release because lexicon-based conventions cannot yet transcribe them consistently. Since classifier predicates are central to symptom description, this removes some of the most clinically relevant material; annotating it is a priority for future releases. Relatedly, fingerspelled items are retained as single opaque gloss units, which our error analysis identifies as a weakness for exactly the content where errors matter most, such as medication names and identifiers.

Finally, evaluation is automatic only. No human evaluation with Deaf users, interpreters, or clinicians was conducted, and automatic metrics under-penalise clinically critical errors such as laterality mistakes or corrupted contact information; the systems reported here are research baselines, not deployable clinical tools.

10 Ethical Statement

The procedure performed in this study was conducted in accordance with the ethical standards of the institutional research committee of our institution. Written informed consent was obtained from all 11 participating signers prior to recording, covering both their participation in the study and the public release of their identifiable video recordings as part of the dataset; participants were informed that, unlike text or audio, signed video cannot be anonymised without destroying its linguistic content, and retained the right to withdraw at any point before release.

Regarding representativeness, the signer pool was balanced by age and gender (Table 2) but is homogeneous in ethnicity, with all 11 signers being white/Caucasian. This reflects the demographic composition of the Catalan Deaf signing community from which participants were recruited, rather than a sampling choice, but it means the dataset does not capture the visual or linguistic variation associated with more ethnically diverse Deaf communities, and systems trained on it should not be assumed to generalise to signers outside this demographic profile without further validation.

Acknowledgments

This work/research has been promoted and financed by the Government of Catalonia through the Aina Project. This work is also supported by MLLM4TRA (PID2024-158157OB-C32) funded by MCIN/AEI/10.13039/501100011033/FEDER, UE. Carlos Escolano is funded by Grant LLM4SL (PID2024-157855OA-C33) MICIU/AEI/10.13039/501100011033/FEDER, UE. Gerard Sant is funded by the SIGMA project (grant no. G-95017-01-07), supported by the Digital Society Initiative (DSI) at the University of Zurich. We warmly thank Delfina Aliaga and Mireia Isal for their work on the annotation and translation processes, as well as the deaf LSC participants who participated in the video recordings, without whom this research would not have been possible.

References

Gemma Barberà, Eulàlia Ribera, and Cristina Gelpí. 2018. Recursos lingüístics i educatius de la llengua de signes catalana (LSC). *Llengua, Societat i Comunicació*, 16:16–25.

Gemma Barberà, Josep Quer, and Santiago Frigola. 2015. Primers passos cap a la documentació de discurs signat. El projecte pilot de constitució del corpus de la llengua de signes catalana [First steps towards the documentation of signed discourse. The pilot project for the creation of the Catalan Sign Language corpus]. *Treballs de Sociolingüística Catalana*, 25:287–302.

Danielle Bragg, Oscar Koller, Mary Bellard, Larwan Berke, Patrick Boudreault, Annelies Braffort, Naomi Caselli, Matt Huenerfauth, Hernisa Kacorri, Tessa

Verhoef, Christian Vogler, and Meredith Ringel Morris. 2019. Sign language recognition, generation, and translation: An interdisciplinary perspective. In *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '19, page 16–31, New York, NY, USA. Association for Computing Machinery.

Necati Cihan Camgöz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 7784–7793. Computer Vision Foundation / IEEE Computer Society.

Yutong Chen, Ronglai Zuo, Fangyun Wei, Yu Wu, Shujie Liu, and Brian Mak. 2022. Two-stream network for sign language recognition and translation. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

Amanda Cardoso Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giró-i-Nieto. 2021. How2sign: A large-scale multimodal dataset for continuous american sign language. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 2735–2744. Computer Vision Foundation / IEEE.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Michael Auli, and Armand Joulin. 2021. Beyond english-centric multilingual machine translation. *J. Mach. Learn. Res.*, 22:107:1–107:48.

Jordan Fenlon and Julie A. Hochgesang, editors. 2022. *Signed Language Corpora*. Gallaudet University Press.

Jens Forster, Christoph Schmidt, Oscar Koller, Martin Bellgardt, and Hermann Ney. 2014. Extensions of the sign language recognition and translation corpus rwth-phoenix-weather. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014*, pages 1911–1916. European Language Resources Association (ELRA).

Alex Graves, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, volume 148 of *ACM International Conference Proceeding Series*, pages 369–376. ACM.

Jie Huang, Wengang Zhou, Qilin Zhang, Houqiang Li, and Weiping Li. 2018. Video-based sign language recognition without temporal segmentation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 2257–2264. AAAI Press.

Institut d'Estudis Catalans. 2025. Corpus de referència de la llengua de signes catalana (LSC) (CORPUS LSC).

Mathias Müller, Zifan Jiang, Amit Moryossef, Annette Rios, and Sarah Ebling. 2023. Considerations for meaningful sign language machine translation based on glosses. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 682–693, Toronto, Canada. Association for Computational Linguistics.

Eva Ostertagova, Oskar Ostertag, and Jozef Kováč. 2014. Methodology and application of the kruskal-wallis test. *Applied Mechanics and Materials*, 611:115–120.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, and 2 others. 2019. *PyTorch: an imperative style, high-performance deep learning library*. Curran Associates Inc., Red Hook, NY, USA.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Josep Quer. 2012. Legal pathways to the recognition of sign languages: A comparison of the catalan and

spanish sign language acts. *Sign Language Studies*, 12(4):565–582.

Josep Quer. 2017. Els projectes de codificació de la llengua de signes catalana (LSC). *Estudis Romànics*, 39:445–452.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Razieh Rastgoo, Kourosh Kiani, and Sergio Escalera. 2021. Sign language recognition: A deep survey. *Expert Syst. Appl.*, 164:113794.

Ricardo Rei, Marcos Treviso, Nuno M. Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José G. C. de Souza, Taisiya Glushkova, Duarte Alves, Luisa Coheur, Alon Lavie, and André F. T. Martins. 2022. CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 634–645, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Gerard Sant, Zifan Jiang, Carlos Escolano, Amit Moryossef, Mathias Müller, Rico Sennrich, and Sarah Ebling. 2025a. Multimodalhugs: Enabling sign language processing in hugging face. *CoRR*, abs/2509.09729.

Gerard Sant, Amit Moryossef, Mathias Müller, and Sarah Ebling. 2025b. Modality matters: Training and tokenization effects in sign-to-text translation. In *Adjunct Proceedings of the 25th ACM International Conference on Intelligent Virtual Agents, IVA Adjunct 2025, Berlin, Germany, September 16-19, 2025*, pages 6:1–6:8. ACM.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

Lars St»hle and Svante Wold. 1989. Analysis of variance (anova). *Chemometrics and Intelligent Laboratory Systems*, 6(4):259–272.

Garrett Tanzer and Biao Zhang. 2025. Youtube-sl-25: A large-scale, open-domain multilingual sign language parallel corpus. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Biao Zhang, Garrett Tanzer, and Orhan Firat. 2024. Scaling sign language translation. In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Benjia Zhou, Zhigang Chen, Albert Clapés, Jun Wan, Yanyan Liang, Sergio Escalera, Zhen Lei, and Du Zhang. 2023. Gloss-free sign language translation: Improving from visual-language pretraining. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 20814–20824. IEEE.

A Training Hyperparameters

Hyperparameter	Value
Trainable components	All model components
Projected feature dimension	512
Optimizer	AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.998$)
LR schedule	Inverse square root
Effective batch size	64 (1×64 accumulation steps)
Gradient clipping	Disabled (max grad norm = 0)
Weight decay	0
Label smoothing	0.1
Frame stride	2
Precision	FP16
Early stopping	Patience = 15 (metric: SacreBLEU)
Decoding metric	SacreBLEU + ChrF

Table 5: Hyperparameter configurations for the End-to-End (E2E) translation model.

B Length and Performance Comparison

Component	Hyperparameter	Value
Visual Backbone (shared)		
CLIP encoder	Checkpoint	`openai/clip-vit-base-patch32`
CLIP encoder	Output representation	512-dim contrastive projection
Temporal CNN (Video-to-Gloss)		
Conv layer 1	Filters	2,048
Conv layer 1	Kernel size	5
Conv layer 1	Stride	2
Conv layer 1	Padding	Same-style
Conv layer 2	Filters	512
Conv layer 2	Kernel size	5
Conv layer 2	Dilation	2 (effective receptive field: 9 frames)
Conv layer 2	Stride	1
Conv layer 2	Output dimension	512
Both layers	Normalisation	Batch normalisation
Both layers	Activation	ReLU
Both layers	Dropout	0.2
Temporal Transformer Encoder (Video-to-Gloss)		
Positional encoding	Type	Sinusoidal
Positional encoding	Dropout	0.1
Encoder	Layers	6
Encoder	Model dimension	512
Encoder	Attention heads	8
Encoder	FFN dimension	2,048
Encoder	Attention dropout	0.1
Encoder	FFN dropout	0.1
Output Head (Video-to-Gloss)		
Linear head	Input dimension	512
Linear head	Gloss vocabulary size	3,951 classes
Linear head	Training/inference loss	CTC
Gloss-to-Text (G2T)		
Backbone	Architecture	M2M-100
Backbone	Parameters	418M
Backbone	Shared vocabulary	128,113 tokens
Backbone	Encoder layers	12
Backbone	Decoder layers	12
Backbone	Model dimension	1,024
Backbone	FFN dimension	4,096
Backbone	Special token `__lsc__`	Marks input as LSC gloss sequence

Table 6: Hyperparameter configuration for the Cascade architecture using the 512-dimensional CLIP contrastive projection variant, split into the Video-to-Gloss (V2G) recognition stage and the Gloss-to-Text (G2T) translation stage.

Hyperparameter	Video-to-Gloss (V2G)	Gloss-to-Text (G2T)
Base model / backbone	CLIP ViT-B/32 (unfrozen)	M2M-100 418M (fine-tuned)
Optimizer	AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.998$)	AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.998$)
Learning rate (backbone)	5×10^{-6}	—
Learning rate (other / full)	5×10^{-4}	3×10^{-5}
LR schedule	Linear warmup + cosine decay	Warmup ratio 0.1
Warmup steps / ratio	200 steps	Ratio 0.1
Max epochs	80	15
Batch size (per device)	2	8
Gradient accumulation	32 (effective 64)	4
Weight decay	0.01 (backbone) / 0.05 (full)	0.05
Gradient clipping	5.0	—
Frame stride	2	—
Max sequence length	—	128 tokens
Decoding	—	Beam search ($k = 5$)
Precision	FP16	FP16
Early stopping	Patience 10 (WER)	Patience 5 (SacreBLEU)

Table 7: Hyperparameter configurations for the Cascade model, split in Video-to-Gloss (V2G) and Gloss-to-Text (G2T) model training.

		Cascade		End-to-End		Oracle	
Variable	**Metric**	**Pearson r**	**Spearman ρ**	**Pearson r**	**Spearman ρ**	**Pearson r**	**Spearman ρ**
Gloss Length	BLEU	-0.266^{***}	-0.422^{***}	-0.187^{**}	-0.398^{***}	-0.277^{***}	-0.233^{***}
	ChrF	-0.095 (n.s.)	0.036 (n.s.)	-0.044 (n.s.)	-0.115 (n.s.)	-0.238^{**}	-0.184^{**}
	BERTScore F1	-0.169^{**}	-0.131^{*}	-0.030 (n.s.)	0.006 (n.s.)	-0.320^{***}	-0.273^{***}
	CometKiwi	-0.113 (n.s.)	-0.017 (n.s.)	-0.003 (n.s.)	0.051 (n.s.)	-0.190^{**}	-0.133^{*}
Ref Word Length	BLEU	-0.309^{***}	-0.512^{***}	-0.242^{***}	-0.473^{***}	-0.303^{***}	-0.251^{***}
	ChrF	-0.143^{*}	0.007 (n.s.)	-0.098 (n.s.)	-0.156^{*}	-0.321^{***}	-0.303^{***}
	BERTScore F1	-0.168^{**}	-0.119 (n.s.)	0.023 (n.s.)	0.091 (n.s.)	-0.335^{***}	-0.295^{***}
	CometKiwi	-0.108 (n.s.)	-0.049 (n.s.)	0.005 (n.s.)	0.075 (n.s.)	-0.198^{**}	-0.149^{*}
Pred Word Length	BLEU	-0.161^{**}	-0.166^{**}	0.006 (n.s.)	0.063 (n.s.)	-0.266^{**}	-0.183^{**}
	ChrF	0.066 (n.s.)	0.280^{***}	0.320^{***}	0.631^{***}	-0.228^{*}	-0.159^{*}
	BERTScore F1	-0.069 (n.s.)	-0.008 (n.s.)	-0.052 (n.s.)	0.040 (n.s.)	-0.345^{***}	-0.253^{***}
	CometKiwi	-0.102 (n.s.)	-0.032 (n.s.)	-0.059 (n.s.)	-0.013 (n.s.)	-0.224^{**}	-0.155^{*}

$^{*}\,p < 0.05$, $^{**}\,p < 0.01$, $^{***}\,p < 0.001$, n.s. = not significant.

Table 8: Sample-Level Correlation Analysis between Length Variables and Evaluation Metrics.

C Translation Examples

The examples below focus on the most frequent and clinically important categories: catastrophic hallucination, V2G recognition collapse, partial gloss confusion, structured contact-information errors, and templated over-generation. Green highlighting marks content that correctly matches or preserves the reference, while red highlighting marks the specific error being illustrated.

Topic hallucination. This is the most common error of the gloss-free end-to-end system. Rather than producing a disfluent or partially correct translation, the model frequently generates a fluent, grammatical Catalan sentence about an entirely different clinical topic. Example C.1 illustrates this behaviour.

Example C.1: Topic substitution in the gloss-free system (Sample 1)

Reference gloss: `SI IX1 PARE-2 ULLS-1 UN IX-ULLS-DRETA-1 CATARACTES-4 UN`

Reference: Sí, el meu pare té cataractes a l'ull dret.
("Yes, my father has cataracts in his right eye.")

E2E prediction: Sí, quan vaig amb bicicleta i em ve l'aire de cara em ploren els ulls, sí.
("Yes, when I ride my bike and the wind blows in my face my eyes water, yes.")

Cascade: Sí, el meu pare nom és el meu pare. (BLEU-1 55.6, chrF 0.712)
Oracle: Sí, el meu pare té un ull esquerre amb cataractes, un altre amb el dret. (BLEU-1 46.7, chrF 0.730)

The end-to-end system retains only the discourse marker *"Sí"* and replaces the clinical statement with an unrelated but fluent sentence about watery eyes while cycling. The output remains loosely eye-related, but it does not contain any of the source information about the father, cataracts, or the affected right eye. The cascade hallucinates less because the recognised glosses still preserve the subject, but the missing cataract and eye-location signs lead the G2T model to produce a semantically incorrect sentence. The oracle translation shows a different problem: even with accurate glosses, the text generator can distort laterality, which is clinically significant in medical communication.

Recognition-stage collapse masking a correct G2T stage. A recurring cascade pattern is that the G2T stage performs well when given an accurate gloss sequence, but the full pipeline fails because the upstream V2G model produces an empty or severely corrupted recognition. Example C.2 is the clearest case: the oracle exactly reproduces the reference, while the cascade collapses to a generic filler utterance.

Example C.2: V2G recognition collapse (Sample 35)

Reference gloss: `DORMIR PROBLEMA HAVER-NO HAVER-NO RES-5 RES-5`
Predicted gloss: `(empty)`

Reference: No tinc cap problema per dormir.
("I have no problem sleeping.")

E2E prediction: Em fa mal el cul, a les vértebres de baix.
("My bottom hurts, in the lower vertebrae.")

Cascade: Sí, sí. (BLEU-1 0.0, chrF 0.103)
Oracle: No tinc cap problema per dormir. (BLEU-1 100.0, chrF 1.000, exact match)

Here the V2G model recognises no glosses at all for a six-token reference sequence. The G2T model therefore receives no meaningful source content and falls back to *"Sí, sí."*, producing simultaneous hallucination, omission, and under-generation. The end-to-end system also fails, but in a different way: it hallucinates an unrelated complaint about lower-back pain.

Example C.3 shows a less extreme version of the same phenomenon. The V2G output preserves the falling/stairs predicate, but substitutes the shopping-centre context with unrelated signs, causing the final translation to combine a correct event frame with invented details.

Example C.3: Partial gloss confusion (Sample 97)

Reference gloss: `IX1-5 CENTRE COMERCIAL CL-2D BAIXANT-ESCALA CL-2 PERSONA-CAIENT-A-LA-VOLTA`
Predicted gloss: `PATINET DUTXAR-SE IX-LLOC ESCALA CL-B CL-2 PERSONA-CAIENT`

Reference: Vaig caure per les escales del centre comercial.
("I fell down the stairs of the shopping centre.")

E2E prediction: Em fa mal el dit gros del peu esquerre, perquè em vaig donar un cop al peu amb la vorera.
("My big toe hurts, because I hit my foot on the curb.")

Cascade: M'he dutxat amb patinet i he caigut per les escales. (BLEU-1 20.0, chrF 0.720)
Oracle: Vaig caure per les escales del centre comercial. (BLEU-1 100.0, chrF 1.000, exact match)

The recognised gloss sequence keeps `PERSONA-CAIENT` and `ESCALA`, so the cascade correctly produces the notion of falling on stairs. However, `CENTRE COMERCIAL` is replaced by `PATINET DUTXAR-SE`, leading to a fluent but impossible clause about showering with a scooter. This type of output is particularly risky because it can remain plausible as a medical complaint while being factually wrong.

Numeric and contact-information errors. Personal contact information is encoded through fingerspelling and digit signs, placing strong demands on temporal resolution and fine-grained handshape recognition. Example C.4 shows how a small recognition error can have a large semantic cost.

Example C.4: Fingerspelled email address (Sample 2)

Reference gloss: `IX1 MAIL VINT I VUIT DC A-N-N-A CINC ARROVA HOTMAIL DC C-O-M`
Predicted gloss: `IX1 MAIL DC A-N-N-A CINC DC PUNT-0E DC C-O-M`

Reference: El meu correu és vint-i-vuit anna cinc arrova hotmail com.
("My email is twenty-eight anna five at hotmail com.")

E2E prediction: El meu correu és EAP, ara no ho sé, EAP asterisc al mig, ni a dalt ni a baix, al mig, vint-i-quatre arrova gmail com.

Cascade: El meu correu és anna cinc punt com. (BLEU-1 70.0, chrF 0.745)
Oracle: El meu correu és vint-i-vuit, anna, cinc, arrova hotmail com. (BLEU-1 70.0, chrF 0.975)

The V2G model drops the leading numeral sequence `VINT I VUIT` and misrepresents the domain marker, so the cascade preserves the broad email template while losing crucial identifier content. The BLEU-1 score remains relatively high because many surrounding words overlap, but a missing two-digit element in contact information is much more serious than the metric suggests.

Templated hallucination and over-generation. The most severe over-generation cases reveal a distinct failure mode in which the end-to-end model appears to fall back on a long memorised response largely independent of the input. Example C.5 shows the most extreme instance.

Example C.5: Templated over-generation (Sample 40)

Reference gloss: `IX-CARA-5 SI`

Reference: Sí, uf.
("Yes, ugh.")

E2E prediction: Sí, em van aplicar calor per poder fer els estiraments, em van aplicar aquesta calor als braços i cames. Però quan vaig començar a jugar a futbol, estava corrent i vaig notar que se m'enrampava la cama, em feia mal i vaig agafar gel, quan vaig aplicar el fred ja em va marxar la rampa, però com que em costava caminar al final vaig seure per descansar.
(A 56-word fabricated narrative about sports injury, heat therapy, and cramping.)

Cascade: Sí, sí. (BLEU-1 50.0, chrF 0.714)
Oracle: Sí, a la cara. (BLEU-1 25.0, chrF 0.476)

Metric-Learning Retrieval for Cross-Signer Isolated Sign Language Recognition

Md. Asif Ali, Md. Mosharaf Hossain Apurbo, Kazi Reyazul Hasan*
Department of Computer Science & Engineering
Bangladesh University of Engineering & Technology, Dhaka, Bangladesh
ijsasif45@gmail.com, mosharaf.apurbo57@gmail.com,
kazireyazulhasan@gmail.com

Abstract

We address cross-signer Isolated Sign Language Recognition (ISLR) for Indian Sign Language as part of the WSLP-2026 shared task, where most target words have exactly one available training video and a system must recognize a query video performed by a signer never seen for that class during training. We approach this with a metric-learning encoder trained in two phases: pretraining on a large, multi-signer American Sign Language corpus to learn general signer-invariance, then fine-tuning on the target vocabulary using every real cross-signer recording we can responsibly source. Our preprocessing detects and extracts multiple performances of a word within a single recording as separate training samples, a technique not previously reported in SLR literature, and adapts a published missing-keypoint interpolation technique into a retrieval rather than classification pipeline. Our final system achieves 67.11% Top-1, 69.20% Top-5, and 70.34% Top-10 accuracy on the WSLP-2026 test-phase evaluation.

1 Introduction

Sign language recognition (SLR) is a key technology for reducing communication barriers for deaf and hard-of-hearing communities, yet the vast majority of SLR research and resources concentrate on a small number of well-resourced sign languages, chiefly American Sign Language (ASL), leaving languages such as Indian Sign Language (ISL) comparatively underserved despite large signing populations. Sign languages are also inherently multimodal, combining hand configuration, body posture, and motion dynamics, and highly variable across signers in speed, articulation, and style, making automatic recognition a challenging visual sequence modeling problem even where resources are plentiful.

In this paper, we address Isolated Sign Language Recognition (ISLR) for ISL as part of the WSLP-2026 shared task: given a query video, recognize which word from the task vocabulary it depicts.

Two central challenges characterize this setting:

- **Extreme per-class scarcity**: for the WSLP-2026 vocabulary, the large majority of target words have exactly one available training video, ruling out the many-examples-per-class training most SLR systems assume.

- **Cross-signer generalization**: a system must recognize a query video performed by a signer whose body proportions, signing speed, and style it has never observed for that class, distinct from the within-signer or seen-signer evaluation common in prior SLR work.

We address these challenges with a metric-learning encoder trained in two phases: pretraining on a large, multi-signer American Sign Language corpus to learn general signer-invariance, then fine-tuning on the target vocabulary using every real cross-signer recording we can responsibly source. Because most classes provide only one training example, we frame recognition as nearest-neighbor retrieval in a learned embedding space rather than closed-set classification (§3).

Our preprocessing pipeline detects and extracts multiple performances of a word within a single dictionary-style recording as separate training samples, a technique not previously reported in SLR literature to our knowledge (§3.2), and adapts a published missing-keypoint interpolation technique, validated only for classification on a different landmark layout, into our retrieval pipeline (§3.2).

*Corresponding author.

Proceedings of the Second Workshop on Sign Language Processing, pages 70–76
October 29, 2026 ©2026 Association for Computational Linguistics

2 Related Work

Sign language datasets. Word-level SLR has been driven substantially by American Sign Language resources: Purdue RVL-SLLL ASL (Wilbur and Kak, 2006), WLASL (Li et al., 2020), and MS-ASL (Vaezi Joze and Koller, 2019), alongside language-specific corpora such as LSA64 for Argentinian Sign Language (Ronchetti et al., 2023) and DEVISIGN for Chinese Sign Language (Chai et al., 2014). For Indian Sign Language specifically, INCLUDE (Sridhar et al., 2020) and CISLR (Joshi et al., 2022) are available word-level resources.

Prior work on this shared task. WSLP's inaugural edition (WSLP-AACL-2025) hosted a word-level recognition task of matching scale (4,398 videos), with reported per-class scarcity consistent with the task we address (Reddy and Kamakshi, 2025 report 80% of classes have at most 2 samples, median exactly 1). Reddy and Kamakshi (2025) establish a closed-set classification baseline: MediaPipe landmarks (Lugaresi et al., 2019) (75 points, no face) into a dilated-convolution temporal convolutional network, reporting 54% Top-1 / 78% Top-5 on their own 3,517/879 train/validation split. The different split and closed-set (rather than retrieval-based) formulation mean the numbers are not directly comparable.

Architectures. Early SLR work relied on CNNs (Tran et al., 2015) and, given sign language's inherently temporal structure, LSTMs (Sams et al., 2023; Khartheesvar et al., 2023; Liao et al., 2019); graph neural networks (Yu et al., 2018) and inflated 3D convolutions (Carreira and Zisserman, 2017) model spatial skeleton structure and spatiotemporal video features respectively. Transformer architectures (Vaswani et al., 2017), despite their success elsewhere, remain comparatively under-explored in SLR, with some recent applications showing promising results (Camgoz et al., 2020; Boháček and Hrúz, 2022). Our encoder follows this transformer lineage, adapting a per-landmark-group masked embedding concept (Hasan and Adnan, 2025): independently embedding each landmark group. Ensembling has also seen limited application in this space (Hrůz et al., 2022): one approach uses majority voting across independently-trained models (Hasan and Adnan, 2025), which suits closed-set classification but does not transfer directly to the open-vocabulary retrieval setting we address; we discuss this design choice in §3.5.

Source	Videos	Classes
WSLP task videos	4,398	4,356
CISLR `test.csv`	1,723	968
CISLR `prototype.csv`	2,018	2,017
Google ASL Signs (GISLR)[1]	94,477	250

Table 1: Training data sources in the final system.

Per-class scarcity and cross-signer generalization. The specific combination we address (as few as one training example per class, generalization required to an unseen signer) is comparatively rare as an explicit evaluation setting in prior SLR work, which more commonly evaluates within-signer or many-examples-per-class regimes. Our repetition-mining contribution (§3.2) is related to, but distinct from, sliding-window multi-crop sampling from long clips (which is blind to performance boundaries rather than aligned to them) and to sign-spotting / dictionary alignment work (Momeni et al., 2020), which aligns dictionary and continuous signing but does not, to our knowledge, convert detected intra-video repetitions into additional training supervision with an ablation demonstrating their value.

3 Method

We address cross-signer Isolated Sign Language Recognition (ISLR) for Indian Sign Language (ISL): given exactly one training example for most target words, a system must recognize a query video of that word performed by a signer never seen for that class during training. This setting differs from standard closed-set sign classification in that the model cannot rely on having observed diverse performers of every class; it must instead learn a general notion of signer-invariance and apply it zero-shot to unseen classes and signers at test time. Figure 1 gives an overview of the full pipeline.

3.1 Data Sources

Our training pool combines the shared task's own data with three external cross-signer sources, summarized in Table 1.

The WSLP task's own training data provides almost exclusively *one video per class*: the core difficulty of the task. **CISLR** (Joshi et al., 2022),

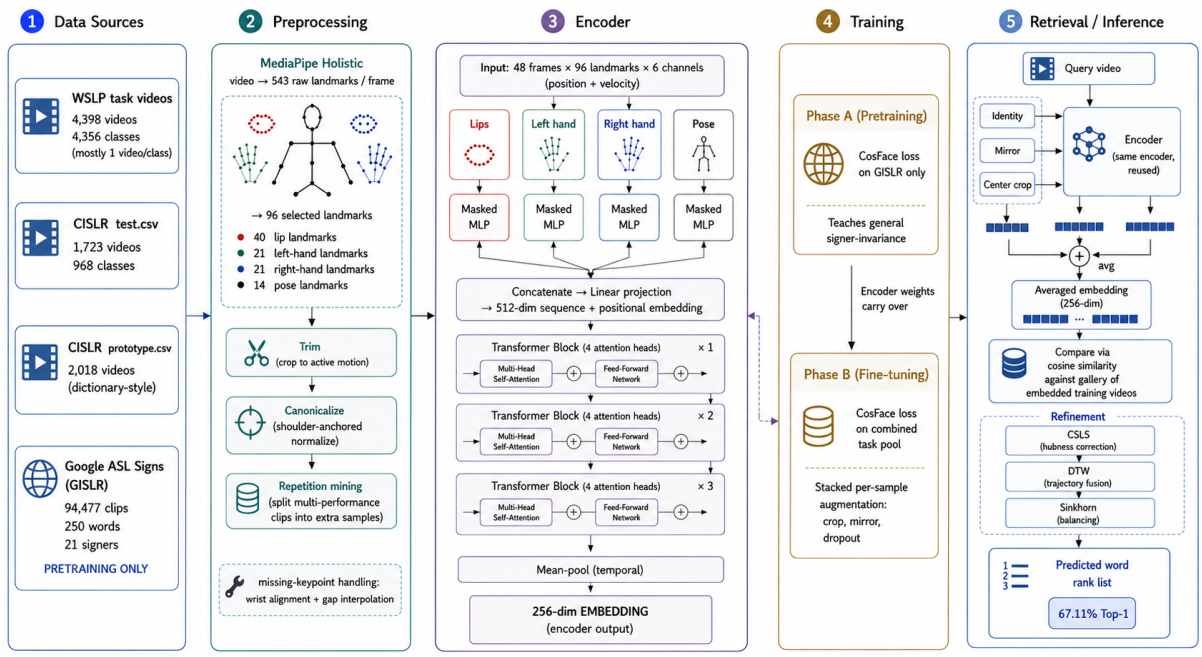


Figure 1: Overview of the pipeline: external cross-signer sources augment the task's own data (§3.1); preprocessing and a per-group masked Transformer encoder (§3.2–§3.3) produce embeddings trained via two CosFace phases (§3.4); at inference, an averaged query embedding is compared against the training gallery and refined via CSLS, DTW, and Sinkhorn balancing (§3.5).

an Indian Sign Language corpus, supplies two components. `test.csv` contributes extra-signer recordings that happen to overlap the task vocabulary. `prototype.csv`, a large (4,765-video) dictionary-style set, is vocabulary-matched to contribute 2,018 videos, 1,049 of them for classes that otherwise had only a single training video (§5.2).

Google ASL Signs (GISLR) is the Kaggle competition training release of the PopSign ASL corpus (Starner et al., 2023) (250 words, 21 signers, ∼94,477 clips), used exclusively for pretraining (§3.4). Although it is a different sign language, it provides the one thing our task data structurally lacks: many distinct signers performing the same word, which is what makes learning general signer-invariance possible before ever seeing the target vocabulary.

3.2 Preprocessing

Each video is processed through MediaPipe Holistic (Lugaresi et al., 2019) to 543 raw landmarks per frame, from which we select 96 (40 lip, 21 left-hand, 21 right-hand, 14 pose including shoulders and wrists). Three preprocessing steps were retained after a component ablation (§5.1) that tested five candidate steps in isolation and removed-from-the-full-recipe:

- **Trim**: crop each clip to its active motion span via a motion-energy detector, removing idle padding before and after the sign.
- **Canonicalize**: shoulder-anchored translation and scale normalization, removing camera-distance and body-size variation across recordings.
- **Repetition mining**: some dictionary-style source videos contain the signer performing the target word more than once. We detect repeated bursts via the same motion-energy signal and extract each as an additional training sample, effectively mining extra supervision from a single recording.

A preprocessing-time mirroring step and a DTW-based consistency filter on mined repetitions were both tested and dropped; both were harmful in isolation and their removal measurably improved the full recipe (§5.1).

Missing-keypoint handling. MediaPipe frequently fails to detect a hand for several consecutive frames, and its independently-estimated pose-model wrist and hand-model wrist positions do not always agree. We adopt a published interpolation and alignment technique (Hasan and

Adnan, 2025): (i) whenever a hand is detected, the pose model's wrist estimate and three adjacent landmarks (the pinky, index, and middle finger bases) are each replaced by the hand model's own corresponding position (their homogeneous-matrix formulation, algebraically equivalent to direct assignment); (ii) for an interior span of frames where a hand is missing but bounded by valid frames on both sides, each of the 21 hand landmarks is linearly interpolated between the two boundary frames. We re-implemented this technique against our own 96-landmark layout and independently verified it against real data before integration (correct target column indices, exact reproduction of the published interpolation formula, and a real test case with 38 missing frames reduced to 1 unrecoverable frame (a leading/trailing gap with no valid frame on one side, which the technique does not cover by design).

3.3 Encoder Architecture

The encoder embeds each 48-frame, 96-landmark, 6-channel (position + masked velocity) clip into a 256-dimensional representation. Each of four landmark groups (lips, left hand, right hand, pose) is independently embedded by a masked two-layer MLP: if a group is entirely absent in a frame, its output is zeroed rather than passed through the network, preserving an explicit missingness signal rather than treating a zero-filled input as informative. A published alternative (Hasan and Adnan, 2025) instead learns a trainable embedding for missing groups; our four group embeddings are also combined by concatenation and linear projection to a 512-dimensional sequence, rather than their learned weighted sum. The sequence is added to a learned positional embedding and passed through 3 Transformer blocks (4 attention heads, MLP ratio 2, dropout 0.1) using standard scaled dot-product multi-head attention (Vaswani et al., 2017). The sequence is mean-pooled and projected to the final 256-dimensional embedding. No classification head is attached to the encoder; training and inference both operate purely in embedding space via the metric-learning objective described next.

3.4 Training

Training proceeds in two phases.

Phase A (pretraining). We pretrain with a CosFace (Wang et al., 2018) metric-learning objective on GISLR (§3.1), which pushes different signers' performances of the same word toward the same region of embedding space. Two of GISLR's 21 signers are held out entirely, giving a genuine cross-signer retrieval check during development. 30 epochs, batch size 1,024, learning rate 1×10^{-4}, CosFace scale 30 / margin 0.20.

Phase B (fine-tuning). The encoder is then fine-tuned with the same CosFace objective on the task's own combined training pool (§3.1–3.2), up to 15 stages of 4 epochs each, batch size 512, learning rate 1×10^{-4}, CosFace scale 30 / margin 0.30. Each training sample additionally receives per-sample augmentation: random temporal crop (0.70–1.0 of clip length), mirror (25% probability), and frame dropout (8%), stacked; a fourth candidate, affine jitter, is disabled in the best configuration.

3.5 Retrieval

At inference, each query is embedded with test-time augmentation (identity, mirror, and a center temporal crop, averaged). Similarity to the training gallery is computed by cosine distance, aggregated per class by taking the maximum similarity over that class's gallery samples, then refined by CSLS hubness correction (Conneau et al., 2018) (k=10), a dynamic-time-warping trajectory-fusion signal on wrist trajectories (weight 0.3), and Sinkhorn transductive balancing (Cuturi, 2013). Isolated testing of Sinkhorn earlier in development found it harmful; in the final configuration, combined with CSLS and DTW fusion, it is genuinely part of the best result; we report this as an interaction effect rather than a reversal (§5).

4 Experiments

We evaluate on the WSLP-2026 shared task's own splits: 4,398 training videos (4,356 unique classes), a 109-video validation set (scored during a training phase), and a 526-video test set (scored during the subsequent, final test phase; this is the result reported throughout §5). The organizers' evaluation reports Top-1, Top-5, and Top-10 retrieval accuracy.

5 Results

Our final system achieves **67.11% Top-1**, 69.20% Top-5, and 70.34% Top-10 on the WSLP-2026 test-phase evaluation.

Configuration	Top-1
Canonicalize + Trim + Rep. mining + Mirror + DTW filter	50.46
Canonicalize + Trim + Rep. mining + DTW filter	52.29 (+1.83)
Canonicalize + Trim + Rep. mining	53.21 (+0.92)
Canonicalize + Rep. mining	52.29 (−0.92)

Table 2: Sequential removal of Mirror and the DTW consistency filter from the full preprocessing recipe, then Trim as a confirmation check (validation-phase scores). Canonicalize + Trim + Rep. mining (row 3) is the winning configuration.

Configuration	Top-1
Baseline (task + CISLR `test.csv`)	53.21
+ CISLR prototype extras	68.81 (+15.60)
+ longer training, no affine jitter	71.56* (+2.75)

Table 3: Cumulative effect of adding CISLR prototype extras (§3.1), validation-phase scores. *Two simultaneous changes (Phase B stage count and disabling one augmentation operation); the combined effect is confirmed, the individual attribution between the two is not.

5.1 Component Ablation

We tested five candidate preprocessing steps (§3.2) on an earlier checkpoint of the system (53.21% Top-1 baseline era; relative effects are what matter here, not the absolute score, which the later data additions in §5.2 substantially improved). Canonicalize and repetition mining were each confirmed positive in isolation (canonicalization accounts for the large majority of the full pipeline's value; repetition mining alone adds +1.84 points) and are held constant throughout Table 2, which traces the sequential removal of the remaining three candidates from the full recipe.

Two results are worth highlighting because the double confirmation (isolated *and* removed-from-recipe) rules out a spurious correlation: trim helps alone (+0.92) and removing it from the winning recipe hurts (the final row above), confirming it is genuinely load-bearing rather than redundant with something else in the recipe. Mirror-time augmentation (−2.75 alone) and the DTW consistency filter (−0.92 alone) both hurt in isolation *and* their removal is exactly what produces the two gains shown above: a consistent, not merely isolated, negative signal.

5.2 Cross-Signer Data Source Ablation

Table 3 shows the effect of adding the CISLR `prototype.csv` dictionary set (§3.1) to the training pool.

Adding the prototype set alone produces a +15.60 point improvement, larger than every subsequently tested component or hyperparameter change combined (§5.1). We verify this is not a leakage artifact: prototype clips are short, single-burst dictionary performances, structurally distinct from the task's own longer, multi-burst recordings (confirmed via an independent motion-energy duration measurement), and a content-level duplicate check (frame-hash comparison across all splits) found zero duplicates.

6 Conclusion

We presented a metric-learning retrieval system for cross-signer Isolated Sign Language Recognition on Indian Sign Language, addressing the WSLP-2026 shared task. A two-phase encoder, pretrained on a multi-signer American Sign Language corpus and fine-tuned on the target vocabulary combined with real cross-signer recordings, achieves 67.11% Top-1, 69.20% Top-5, and 70.34% Top-10 accuracy on the test-phase evaluation. Beyond the final system, we contribute a repetition-mining technique for extracting additional training supervision from dictionary-style recordings, and a cross-paradigm port of a published missing-keypoint interpolation technique into a retrieval setting. Future work could explore further sources of real cross-signer supervision, which this work finds to be the central lever for generalizing to unseen signers under extreme per-class scarcity.

7 Limitations

Several components are not isolated with dedicated ablations on the final configuration. We do not report the final system trained with and without Phase A (ASL) pretraining, nor a breakdown of the retrieval refinement stack (CSLS, DTW fusion, Sinkhorn balancing) into its individual and combined contributions; Section 3.5 reports Sinkhorn's isolated-versus-combined interaction effect as observed during development, not as a controlled ablation on the released system. The missing-keypoint interpolation fix (Section 3.2) is part of the final recipe, but its contribution is confounded with a simultaneous retuning of training hyperparameters, so its standalone effect cannot be separated post hoc.

The preprocessing ablations in Table 2 are measured on a 109-video validation set, where a 0.92-point difference corresponds to a single additional correct prediction, and we do not report results across repeated random seeds. Separately, nominally similar retrained variants of this pipeline showed roughly a 2-point spread in test-phase Top-1 across submissions, so deltas of this size should be read with that variance in mind rather than as precise, seed-independent effects.

We verify that CISLR extras contain no exact content duplicates with the task's own splits (Section 5), but signer identities are not available for either corpus, so we cannot verify that no signer appears in both CISLR extras and the WSLP validation or test splits. Cross-signer generalization results should be read with this caveat.

This paper's contribution is the retrieval formulation, data sources, and preprocessing findings for this task, not an architecture search: we do not compare against modern video-language foundation models or alternative spatiotemporal backbones, and Top-5/Top-10 gains over Top-1 are reported without a retrieval-rank error analysis.

References

Matyas Boháček and Marek Hrúz. 2022. Sign pose-based transformer for word-level sign language recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, pages 182–191.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10023–10033.

Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6299–6308.

Xiujuan Chai, Hanjie Wang, and Xilin Chen. 2014. The devisign large vocabulary of chinese sign language database and baseline evaluations. Technical Report VIPL-TR-14-SLR-001, Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences (CAS).

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *International Conference on Learning Representations (ICLR)*.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems 26 (NIPS 2013)*, pages 2292–2300.

Kazi Reyazul Hasan and Muhammad Abdullah Adnan. 2025. Empath: Mediapipe-aided ensemble learning with attention-based transformers for accurate recognition of bangla word-level sign language. In *Pattern Recognition: 27th International Conference, ICPR 2024, Kolkata, India, December 1–5, 2024, Proceedings, Part XXI*, Lecture Notes in Computer Science, pages 355–371, Cham. Springer Nature Switzerland.

Marek Hrůz, Ivan Gruber, Jakub Kanis, Matyas Boháček, Miloš Hlaváč, and Zdeněk Krňoul. 2022. One model is not enough: Ensembles for isolated sign language recognition. *Sensors*, 22(13):5043.

Abhinav Joshi, Ashwani Bhat, Pradeep S, Priya Gole, Shreyansh Agarwal, Shashwat Gupta, and Ashutosh Modi. 2022. Cislr: Corpus for indian sign language recognition. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10357–10366, Abu Dhabi, United Arab Emirates.

G Khartheesvar, Mohit Kumar, Anil Kumar Yadav, and Divakar Yadav. 2023. Automatic indian sign language recognition using mediapipe holistic and lstm network. *Multimedia Tools and Applications*, 83(20):58329–58348.

Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. 2020. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1459–1469.

Yanqiu Liao, Pengwen Xiong, Weidong Min, Weiqiong Min, and Jiahao Lu. 2019. Dynamic sign language recognition based on video sequence with blstm-3d residual networks. *IEEE Access*, 7:38044–38054.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*.

Liliane Momeni, Gül Varol, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. 2020. Watch, read and lookup: Learning to spot signs from multiple supervisors. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 291–308.

Tatigunta Bhavi Teja Reddy and Vidhya Kamakshi. 2025. Pose-based temporal convolutional networks for isolated indian sign language word recognition. In *Proceedings of the Workshop on Sign Language Processing (WSLP)*, pages 47–50, December 20-24,

2025. Association for Computational Linguistics. WSLP-AACL-2025; National Institute of Technology Calicut.

Franco Ronchetti, Facundo M Quiroga, C Estrebou, Laura Lanzarini, and A Rosete. 2023. Lsa64: An argentinian sign language dataset. *arXiv preprint arXiv:2310.17429*.

Abu Sams, Ashraf Hasan Akash, and Sejuti Mahruba Rahman. 2023. Signbd-word: Video-based bangla word-level sign language and pose translation. In *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pages 1–7.

Advaith Sridhar, Rohith Gandhi Ganesan, Pratyush Kumar, and Mitesh Khapra. 2020. Include: A large scale dataset for indian sign language recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1366–1375.

Thad Starner, Sean Forbes, Matthew So, David R. Martin, Rohit Sridhar, Gururaj Deshpande, Sam Sepah, Sahir Shahryar, Khushi Bhardwaj, Tyler Kwok, Daksh Sehgal, Saad Hassan, Bill Neubauer, Sofia Vempala, Alec Tan, Jocelyn Heath, Unnathi Kumar, Priyanka Mosur, Tavenner Hall, and 5 others. 2023. Popsign asl v1.0: An isolated american sign language dataset collected via smartphones. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023), Datasets and Benchmarks Track*.

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4489–4497.

Hamid Reza Vaezi Joze and Oscar Koller. 2019. Ms-asl: A large-scale data set and benchmark for understanding american sign language. In *British Machine Vision Conference (BMVC)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. 2018. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5265–5274.

Ronnie Wilbur and Avinash C. Kak. 2006. Purdue rvl-slll american sign language database. Technical report, Purdue University.

Bing Yu, Haoteng Yin, and Zhanxing Zhu. 2018. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3634–3640. Cited here as the ST-GCN methodological reference, following EMPATH's usage – not an SLR-specific application.

Multiple-Instance Contrastive Alignment for Pose-Based Sign Spotting

Md. Mosharaf Hossain Apurbo, Md. Asif Ali, Kazi Reyazul Hasan*
Department of Computer Science & Engineering
Bangladesh University of Engineering & Technology, Dhaka, Bangladesh
mosharaf.apurbo57@gmail.com, ijsasif45@gmail.com,
kazireyazulhasan@gmail.com

Abstract

Sign spotting, the task of deciding whether an isolated query sign occurs inside a clip of continuous signing, enables sign language video retrieval without gloss or text annotation. We describe our system for the WSLP@EMNLP-2026 Word/Sign Presence Prediction shared task on Indian Sign Language: a lightweight (2.86M-parameter) pose-based Siamese encoder that embeds both clips in one space, forms an explicit frame-to-frame cosine similarity map, and reduces it with multiple-instance pooling over temporal windows. Training combines binary cross-entropy with a multiple-instance contrastive objective (MIL-NCE). The system reaches 72.69 accuracy and 71.95 macro-F1 on the blind test set, against 61.66 and 58.42 for the published pose baseline on the same data. Ablations identify contrastive alignment as the component that makes the architecture work: the MIL-NCE objective is what places the two clips in a comparable metric space, and training does not leave chance without it. We further document a label-balance property of the released splits that fixes 76.3% of blind test labels without a model, and recommend a change for future editions.

1 Introduction

Most sign language processing targets either *isolated* recognition, which classifies a pre-segmented sign against a fixed vocabulary, or *continuous* recognition and translation, which maps a full clip to gloss or text (Koller et al., 2015; Camgoz et al., 2020). Neither supports the operation an archive search actually needs: deciding whether a particular sign occurs somewhere inside unsegmented signing. That operation is **sign spotting** (Momeni et al., 2020; Jiang et al., 2021b). We study it in its minimal, gloss-free form under the WSLP@EMNLP-2026 Word/Sign Presence Prediction shared task on Indian Sign Language (ISL), a language whose annotated corpora have emerged only recently (Sridhar et al., 2020; Joshi et al., 2022, 2023). Given an isolated query clip and a continuous sentence clip, the system predicts a binary presence label from 2D pose keypoints alone.

The difficulty is the *citation-to-continuous* gap: a query sign is articulated slowly and deliberately, bounded by rest, whereas the same sign inside a sentence is temporally compressed, spatially reduced and co-articulated with its neighbours (Momeni et al., 2020; Jiang et al., 2021b). The benchmark also closes the usual shortcuts — the positive rate is exactly 0.5 and per-query rates are near chance — so a model that does not genuinely compare its two inputs scores at chance.

Our system places both clips in one metric space with a shared encoder and reads the decision off an explicit frame-to-frame similarity map through multiple-instance pooling. Our contributions are:

1. A 2.86M-parameter Siamese pose encoder with an explicit alignment head, reaching 72.69 accuracy and 71.95 macro-F1 on the blind test set against 61.66 and 58.42 for the published pose baseline (§3, §5).
2. A thirteen-configuration ablation showing that the MIL-NCE objective is the single load-bearing component: without it the model is at chance, and with it every other component we added can be removed, individually within 1.5 accuracy points or all together at no cost (§5).
3. A label-balance property of the released splits that determines 76.3% of blind test labels from the released files alone, with a recommended fix for future editions (§6).

2 Related Work

Sign spotting. Early systems mined sign instances from broadcast footage using subtitles as

*Corresponding author.

Proceedings of the Second Workshop on Sign Language Processing, pages 77–83
October 29, 2026 ©2026 Association for Computational Linguistics

weak supervision — weakly aligned (Buehler et al., 2009) or through subtitle–sign co-occurrence (Pfister et al., 2013) — matching candidates with dynamic time warping (Sakoe and Chiba, 1978). The closest modern line is the Oxford series: Albanie et al. (2020) use mouthing as a weak supervisor, and Momeni et al. (2020) add subtitles and an isolated dictionary, training a shared embedding space with a multiple-instance contrastive loss (MIL-NCE) so a dictionary exemplar lands near the same sign performed inside continuous signing. Varol et al. (2022) scale this dictionary route, Varol et al. (2021) localise signs with attention, and Bull et al. (2021) align subtitles at the sentence level. Jiang et al. (2021b) attack the problem one-shot with a Transformer that reconciles the temporal-scale mismatch between an isolated query and its co-articulated instance. Our system is the "lookup" branch of Momeni et al. (2020), restricted to pose input, a binary presence output and a single GPU. Treating a sentence as a bag of temporal windows that is positive if *any* window matches is a standard MIL formulation (Dietterich et al., 1997; Ilse et al., 2018); the MIL-NCE objective itself follows Miech et al. (2020), an InfoNCE loss (van den Oord et al., 2018) whose positive is a maximum over candidate alignments rather than one fixed pair.

Pose-based sign modelling. Skeleton input is standard for low-resource sign recognition (Jiang et al., 2021a; Hu et al., 2023), and Selvaraj et al. (2022) release pose-based pretrained models covering under-served languages including ISL. We keep the encoder small: no in-domain pretraining corpus is available for this task.

The WSLP shared task. Johnny et al. (2025) describe the system for the 2025 edition. They encode each frame with a 2D CNN over the MediaPipe keypoint grid and pass `[CLS] query [SEP] sentence` through a single BERT-style encoder (Devlin et al., 2019), classifying from the pooled `[CLS]` token. We keep their framing — video-to-video, no gloss, pose rather than RGB — but replace the joint encoder over a concatenated pair with a shared encoder and an explicit alignment map. Their ablations found plain BCE to beat contrastive training; ours find the opposite, and §5 argues the architectural difference is what reconciles the two results.

Block	Dim	Normalisation
Body (15 joints)	30	shoulder-centred, width-scaled
Body velocity	30	first difference
Right hand	42	wrist-centred, RMS-scaled
Left hand	42	wrist-centred, RMS-scaled
Geometry	9	wrist→nose, wrist→wrist
Validity mask	4	pose / RH / LH / face
Face (37 points)	74	nose-anchored, RMS-scaled
Total	**231**	

Table 1: Per-frame feature layout (§3.1).

3 Method

Figure 1 shows the pipeline. A sentence clip of T_s frames and a query clip of T_q frames are converted to normalised pose features, embedded by one shared encoder, and compared through an explicit frame-to-frame cosine similarity map that a multiple-instance head reduces to a presence score.

3.1 Pose Features

We work from 2D keypoints rather than RGB to suppress background, appearance and signer-identity cues (Johnny et al., 2025; Selvaraj et al., 2022). The signer is localised per frame with YOLOv8-pose (Jocher et al., 2023), frames are cropped to a shoulder-centred box, and MediaPipe (Lugaresi et al., 2019) landmarkers give 25 body joints, 21 joints per hand and a 37-point reduced face mesh.

From those coordinates we derive the 231-dimensional descriptor in Table 1. Body joints are centred on the shoulder midpoint and scaled by shoulder width; hands are centred on the wrist and scaled by root-mean-square radius, which isolates handshape from camera distance and body size; the face mesh is nose-anchored and scaled the same way. Undetected articulators are forward- then backward-filled, and a validity mask records which components were imputed. Features are derived on the fly, so augmentation acts on raw coordinates before normalisation.

3.2 Shared Pose Encoder

One encoder embeds both clips into a common d=256 space, so that cosine similarity between a sentence frame and a query frame is meaningful. Each feature block has its own linear stem, with the left- and right-hand stems sharing weights so a handshape is encoded identically on either hand. A residual grouped temporal convolution adds local context, after which we add a learned positional

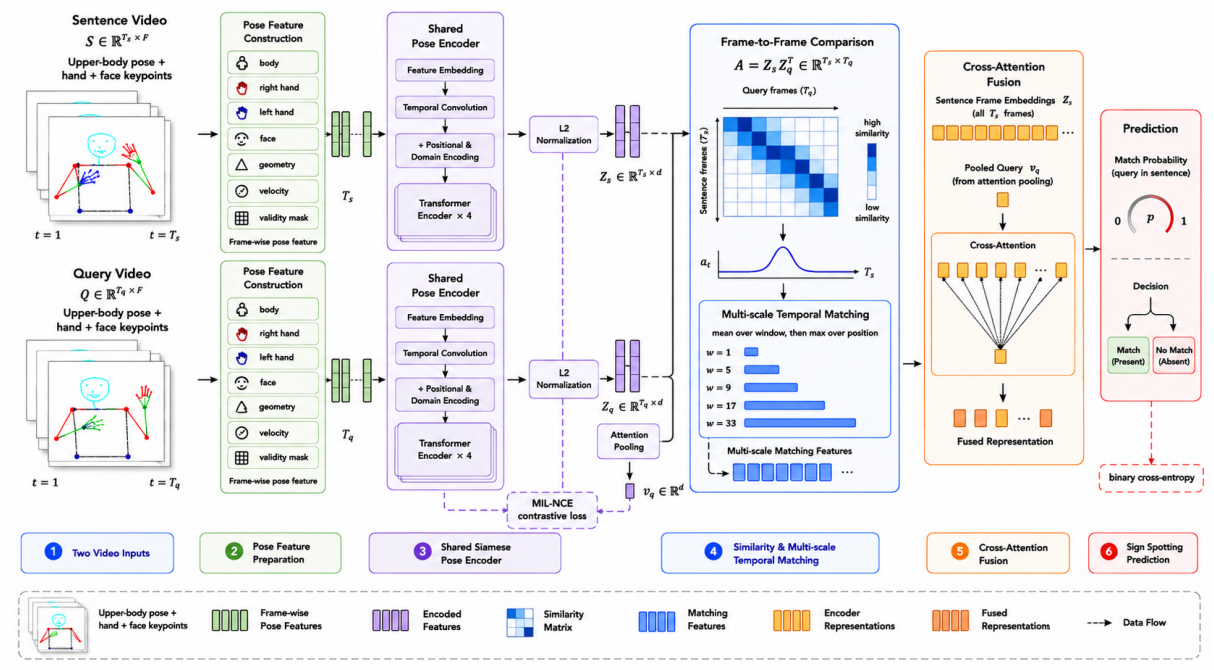


Figure 1: The sign spotting pipeline. One shared (Siamese) encoder maps the continuous sentence clip and the isolated query clip into the same 256-dimensional space, separated only by an additive learned domain embedding. The alignment branch reduces the frame-to-frame cosine similarity map by multiple-instance pooling over temporal windows; the interaction branch cross-attends the pooled query into the sentence frames. Training combines binary cross-entropy with a MIL-NCE contrastive loss applied directly to the embeddings.

embedding and a learned *domain embedding*, $\mathbf{e}_{\text{sent}}$ or $\mathbf{e}_{\text{query}}$. The domain embedding lets the shared trunk absorb systematic framing and tempo differences between citation and co-articulated signing without splitting the geometry into two spaces.

A 4-layer pre-norm Transformer encoder (Vaswani et al., 2017; Xiong et al., 2020) with 8 heads follows. It returns ℓ_2-normalised per-frame embeddings $\mathbf{Z}^s \in \mathbb{R}^{T_s \times d}$ and $\mathbf{Z}^q \in \mathbb{R}^{T_q \times d}$, together with an attention-pooled query vector $v_q \in \mathbb{R}^d$.

3.3 Multi-Scale Multiple-Instance Alignment

We form the frame-to-frame cosine similarity map

$$\mathbf{A} = \mathbf{Z}^s \, \mathbf{Z}^{q\top} \in \mathbb{R}^{T_s \times T_q}, \tag{1}$$

with query padding masked out. Each sentence frame t is scored against its best-matching part of the query by a smooth maximum,

$$a_t = \sum_j \operatorname{softmax}_j\big(\mathbf{A}_{t,:}\big)_j \, A_{tj}, \tag{2}$$

giving an activation trace $\mathbf{a} \in \mathbb{R}^{T_s}$ over sentence frames. Because a sign occupies an unknown number of frames, we pool $\mathbf{a}$ at several temporal scales $w \in \{1, 5, 9, 17, 33\}$ under a multiple-instance reading, in which the sentence is positive if *some* window matches (Momeni et al., 2020; Jiang et al., 2021b):

$$s_w = \max_t \frac{1}{|W_t^w|} \sum_{u \in W_t^w} a_u, \tag{3}$$

where W_t^w is the set of valid frames in the width-w window centred at t. The five scale scores, a top-k mean of $\mathbf{a}$, its mean and the global maximum of $\mathbf{A}$ form an 8-dimensional alignment descriptor.

In parallel, an interaction branch cross-attends v_q into the full sentence frame sequence to obtain a context vector c, and forms the comparison vector $[\,c\,;\, v_q\,;\, c \odot v_q\,;\, c - v_q\,]$. The concatenated alignment and interaction features are passed to a 2-layer MLP that outputs $\hat{y}$, the probability that the query sign occurs in the sentence.

3.4 Training Objective

$$\mathcal{L} = \mathcal{L}_{\text{BCE}} + 0.3\,\mathcal{L}_{\text{NCE}}. \tag{4}$$

The first term is binary cross-entropy on the logit with label smoothing. The second structures the embedding space directly rather than only through the classifier head, following the MIL-NCE objective (Miech et al., 2020; Momeni et al., 2020). For a

batch of B pairs define the sentence-to-query score

$$\sigma_{ij} = \max_{t} \left\langle \mathbf{z}_i^s[t],\ v_{q_j} \right\rangle, \tag{5}$$

which is the multiple-instance step: the alignment position t is a free variable rather than a supervised one. Over the subset $\mathcal{P}$ of rows whose label is positive we take a symmetric InfoNCE loss with temperature $\tau = 0.07$,

$$\mathcal{L}_{\text{NCE}} = \tfrac{1}{2}\Big[\text{CE}_{\mathcal{P}}\big(\sigma/\tau\big) + \text{CE}_{\mathcal{P}}\big(\sigma^{\top}/\tau\big)\Big], \tag{6}$$

where the target of row i is index i itself. Off-diagonal entries that pair a sentence with a duplicate query, or with a query labelled present for that sentence in training, are masked out so valid matches are never pushed apart.

4 Experimental Setup

4.1 Data

The shared task releases 25,432 labelled training pairs over 7,857 sentence clips and 1,410 query clips, exactly balanced between present and absent, plus 1,413 validation and 1,413 test pairs whose labels are withheld. Five training pairs are dropped because two clips failed pose extraction, leaving 25,427.

Two properties of the released splits shaped our protocol. First, the blind splits reuse training clips in new combinations: 95.97% of test sentences and 99.72% of test queries appear in training, but only 2 of 1,413 exact test *pairs* do. Evaluation therefore measures generalisation to new pairings of seen clips, which we match with a **pair-level** cross-validation split, reporting a sentence-grouped split separately as a bound. Second, the negatives are drawn at random rather than mined: a negative query is positive elsewhere in the same source video for only 1.63% of negative pairs, *below* the 3.30% rate from permuting queries at random. Uniformly sampled extra negatives are therefore distributionally close to the released ones.

The leaderboard reports accuracy with macro-averaged precision, recall and F1; we use macro-F1 (F1_{M}) as the headline score throughout.

4.2 Protocol and Implementation

The leaderboard system is a 5-fold stratified pair-level cross-validation ensemble with horizontal-mirror test-time augmentation, using one decision threshold tuned on the pooled out-of-fold predictions and applied unchanged to the blind rows. It also trains on one extra uniformly sampled negative per released pair, redrawn each epoch and rejected if the pair appears in any released file, and applies a gated prior that transfers the label of a near-duplicate query when one exists for the same sentence; the prior fires on 0.2% of rows and we report it for completeness rather than as a contribution. Ablations (§5) use a cheaper protocol: one 80:20 stratified pair-level split, one run per configuration, no ensembling and no synthetic negatives, so the latter is measured in its own row rather than hidden underneath every row.

System	Acc.	F1_{M}	P_{M}	R_{M}
2D CNN + joint enc.†	61.66	58.42	67.16	61.74
Ours	**72.69**	**71.95**	**75.46**	**72.74**

Table 2: Blind test scores (%), macro-averaged. †Johnny et al. (2025), the 2025 edition of this shared task; see the caveat in the text.

The model has 2.86M parameters and is trained for 30 epochs with AdamW (Loshchilov and Hutter, 2019) at learning rate 3×10^{-4}, weight decay 0.02, batch size 32, cosine decay after a short warmup, and an exponential moving average of the weights for evaluation. Sequences are capped at 288 sentence and 128 query frames. Augmentation — mirroring, rotation, scaling, translation, jitter, temporal speed perturbation and articulator dropout — is applied to raw coordinates, with a single mirror decision per *pair* so both clips stay in the same reference frame. The decision threshold maximises positive-class F1 subject to the predicted positive rate staying within $[0.35, 0.65]$, which prevents a weak model from collapsing onto the all-ones solution. One run takes about 30 minutes on a single NVIDIA T4.

5 Results

5.1 Blind Test Result

Table 2 gives our official blind-test score: 72.69 accuracy and 71.95 macro-F1, against 61.66 and 58.42 for Johnny et al. (2025). The two editions release identical split statistics, so the comparison is on matched data, but we could not verify that the test labels are identical across editions and we did not rerun their system; we therefore read Table 2 as a strong indication rather than a controlled comparison. Out-of-fold on the training pairs the ensemble scores 72.37 accuracy and 72.07 macro-F1, within half a point of the blind result, so the

Configuration	Acc.	$F1_M$	P_M	R_M
Full model	72.63	72.28	73.85	72.63
− MIL-NCE loss	**50.92**	**49.84**	**51.01**	**50.92**
− multi-scale MIL	73.12	72.64	74.87	73.12
− cross-attention	72.77	72.57	73.44	72.77
− domain token	73.18	72.83	74.43	73.18
− face keypoints	73.54	73.22	74.70	73.54
− velocity	72.47	71.86	74.60	72.47
− geometry	74.09	73.85	74.97	74.09
− vel. and geom.	72.79	72.34	74.38	72.79
− MIL, domain, face	73.32	72.84	75.09	73.32
− all but MIL-NCE	72.63	72.30	73.75	72.63
− mirror TTA	72.04	71.68	73.23	72.04
+ synth. negatives	72.28	71.95	73.38	72.28
Sentence-level split	69.96	69.40	71.39	69.89

Table 3: Ablations on one 80:20 pair-level split (%, macro-averaged), one run each. "− all but MIL-NCE" removes the face, velocity and geometry blocks, the domain token, the cross-attention branch and the extra MIL scales together. Only removing MIL-NCE (bold) breaks the model. The last row uses a different validation partition and is a generalisation bound, not a comparable ablation.

cross-validation protocol is a reliable proxy here.

5.2 Ablations

MIL-NCE is the whole system. Removing the contrastive term (§3.4) collapses the model to chance, from 72.63 to 50.92 accuracy, and its ranking quality falls with it (AUC $0.83 \rightarrow 0.51$). Nothing else in Table 3 comes close to that effect. The interpretation is that the classifier head reads a cosine similarity map, and BCE alone gives no gradient that makes that map metric; the contrastive term is what puts the two clips in a comparable space in the first place. This also resolves the apparent disagreement with Johnny et al. (2025), who found contrastive supervision unhelpful: in a concatenated single-trunk encoder the decision never passes through an explicit similarity, so a contrastive term is an unrelated auxiliary objective rather than a prerequisite.

Everything else is within noise. The ten remaining removals span 72.04 to 74.09 accuracy, a 2.05-point band around a 72.63 baseline, and seven of them *improve* on it. With one seed and one partition per row we cannot separate these from run-to-run variation, so we claim no effect for any of them individually. The strongest statement the table does support is the − all but MIL-NCE row: removing the face block, velocity, geometry, the domain token, the cross-attention branch and multi-scale pooling all at once — a 157-dimensional feature vector and 2.43M parameters, 15% fewer — matches the full model's accuracy to two decimal places. The components layered on top of the contrastive objective are not earning their place.

Synthetic negatives. Doubling the training set with uniformly sampled negatives costs 0.35 accuracy points and doubles training time. It is in the leaderboard system because that system was trained before this measurement existed; it should be dropped.

Generalisation bound. Under a sentence-grouped split, where no validation sentence is seen in training, the model reaches 69.96 accuracy and 69.40 macro-F1, i.e. 2.7 and 2.9 points below the pair-level run. This row is not comparable to the rest of the table because its validation partition differs; it bounds how much of the headline score depends on the blind splits reusing training clips (§4.1).

6 A Label-Balance Artefact in the Splits

The released files carry a property with a larger effect on leaderboard scores than any modelling choice in this paper: *for every sentence, once the training, validation and test rows are pooled, positives equal negatives.*

Fix a sentence with c labelled training rows of which p are positive, and k blind rows. The property fixes total positives at $(c + k)/2$, so the blind rows must contain $\text{need} = (c + k)/2 - p$ positives; when $\text{need} \in \{0, k\}$ every blind row for that sentence is determined. It holds on all 7,910 sentences, and hiding 1,413 random labelled rows and re-deriving them recovers all of them exactly. Applying it pins 76.3% of test rows and 75.6% of validation rows, so answering the pinned rows and guessing the rest has expected accuracy 0.8815 with no model at all.

Exploiting this would break no rule, but an accuracy produced by counting rows measures nothing about sign language; the failure mode is a familiar one (Poliak et al., 2018; Geirhos et al., 2020). We did not use the property in any submission, and disclosed it to the organisers before submitting this paper. For future editions we recommend assigning blind rows so per-sentence label counts do not sum to a fixed ratio.

7 Conclusion

We presented a 2.86M-parameter pose-based Siamese system for gloss-free sign spotting in Indian Sign Language, which reaches 72.69 accuracy and 71.95 macro-F1 on the WSLP@EMNLP-2026 blind test set. Placing both clips in one embedding space and reading the decision off an explicit frame-to-frame similarity map is what makes the architecture work, and the ablations locate the reason precisely: the MIL-NCE contrastive objective is the only component whose removal matters, while every feature block and architectural addition we layered on top is within run-to-run noise of a 2.43M-parameter model that keeps the contrastive loss and drops the rest. For this task, effort is better spent on the objective that shapes the shared space than on enriching the encoder that fills it.

Limitations

Every ablation row is one run on one seed and one partition. The MIL-NCE effect is far too large to be noise; no other row is, and we draw no conclusion from any of them individually. In that protocol the retained epoch and the decision threshold are both chosen on the partition that is then scored, so those absolute values are optimistic, though the bias applies equally to all rows and the blind test numbers are unaffected. The decoder is also miscalibrated: it predicts positive on 66.4% of test rows where the true rate is near 50%, trading positive precision for recall. Finally, all results come from one ISL corpus, and we could not obtain further blind measurements because the evaluation portal closed before this paper was written, so the single submitted test score is the only blind evidence we can offer.

References

Samuel Albanie, Gül Varol, Liliane Momeni, Triantafyllos Afouras, Joon Son Chung, Neil Fox, and Andrew Zisserman. 2020. BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Patrick Buehler, Mark Everingham, and Andrew Zisserman. 2009. Learning sign language by watching TV (using weakly aligned subtitles). In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hannah Bull, Triantafyllos Afouras, Gül Varol, Samuel Albanie, Liliane Momeni, and Andrew Zisserman. 2021. Aligning subtitles in sign language videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186. Association for Computational Linguistics.

Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. 1997. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1–2):31–71.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Hezhen Hu, Weichao Zhao, Wengang Zhou, and Houqiang Li. 2023. SignBERT+: Hand-model-aware self-supervised pre-training for sign language understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Maximilian Ilse, Jakub Tomczak, and Max Welling. 2018. Attention-based deep multiple instance learning. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*.

Songyao Jiang, Bin Sun, Lichen Wang, Yue Bai, Kunpeng Li, and Yun Fu. 2021a. Skeleton aware multi-modal sign language recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*.

Tao Jiang, Necati Cihan Camgöz, and Richard Bowden. 2021b. Looking for the signs: Identifying isolated sign instances in continuous video footage. In *Proceedings of the British Machine Vision Conference (BMVC)*.

Glenn Jocher, Ayush Chaurasia, and Jing Qiu. 2023. Ultralytics YOLOv8. `https://github.com/ultralytics/ultralytics`.

Samuel Ebimobowei Johnny, Blessed Guda, Emmanuel Enejo Aaron, and Assane Gueye. 2025. Pose-based sign language spotting via an end-to-end encoder architecture. In *Proceedings of the Workshop on Sign Language Processing (WSLP)*, pages 71–76. Association for Computational Linguistics.

Abhinav Joshi, Susmit Agrawal, and Ashutosh Modi. 2023. ISLTranslate: Dataset for translating Indian sign language. In *Findings of the Association for Computational Linguistics: ACL 2023*. Association for Computational Linguistics.

Abhinav Joshi, Ashwani Bhat, Pradeep S, Priya Gole, Shashwat Gupta, Shreyansh Agarwal, and Ashutosh Modi. 2022. CISLR: Corpus for Indian sign language recognition. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Oscar Koller, Jens Forster, and Hermann Ney. 2015. Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. *Computer Vision and Image Understanding*, 141:108–125.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. MediaPipe: A framework for building perception pipelines. *Computing Research Repository*, arXiv:1906.08172.

Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 2020. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Liliane Momeni, Gül Varol, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. 2020. Watch, read and lookup: Learning to spot signs from multiple supervisors. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*.

Tomas Pfister, James Charles, and Andrew Zisserman. 2013. Large-scale learning of sign language by watching tv (using co-occurrences). In *Proceedings of the British Machine Vision Conference (BMVC)*.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 180–191. Association for Computational Linguistics.

Hiroaki Sakoe and Seibi Chiba. 1978. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49.

Prem Selvaraj, Gokul N C, Pratyush Kumar, and Mitesh Khapra. 2022. OpenHands: Making sign language recognition accessible with pose-based pretrained models across languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Advaith Sridhar, Rohith Gandhi Ganesan, Pratyush Kumar, and Mitesh Khapra. 2020. INCLUDE: A large scale dataset for Indian sign language recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *Computing Research Repository*, arXiv:1807.03748.

Gül Varol, Liliane Momeni, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. 2021. Read and attend: Temporal localisation in sign language videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Gül Varol, Liliane Momeni, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. 2022. Scaling up sign spotting through sign language dictionaries. *International Journal of Computer Vision*, 130(6):1416–1439.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. 2020. On layer normalization in the transformer architecture. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*.

Isolated Sign Language Recognition for Icelandic Sign Language: Experiments in a Low-resource Setting

Finnur Ágúst Ingimundarson[1], **Guðný Björk Þorvaldsdóttir**[2]
Mathias Müller[1], **Sarah Ebling**[1]
[1]Department of Computational Linguistics, University of Zurich
[2]Communication Centre for the Deaf and Hard of Hearing in Iceland
finnuragust.ingimundarson@uzh.ch
gudny.bjork.thorvaldsdottir@shh.is
{mmueller,ebling}@cl.uzh.ch

Abstract

We present the first experiments on isolated sign language recognition (ISLR) for Icelandic Sign Language (ÍTM). We use ÍTM SignWiki, a dataset derived from a bilingual Icelandic–ÍTM online dictionary. It is genuinely low-resource: 1,845 videos cover 849 classes, 86% of which have only two examples, making the full task effectively one-shot recognition across signers. We compare two open-source ISLR frameworks, OpenHands and SPOTER, on three tasks of increasing vocabulary size (22, 117 and 849 classes), and evaluate three pose estimators and two forms of cross-lingual transfer. With ÍTM data alone, SPOTER outperforms OpenHands on all three tasks, and MediaPipe poses give better results than AlphaPose or SDPose. Cross-lingual transfer brings the largest gains: pretraining SPOTER on American Sign Language data before finetuning on ÍTM raises accuracy by 14–24 percentage points, to 72.7%, 47.9% and 22.6% on the three tasks, and multilingual training with data from six other sign languages lifts OpenHands from 1.41% to 28.86% on the full task. Although far from practical use, the results suggest that transfer from better-resourced sign languages is promising for very low-resource ones. We release our adapted versions of both frameworks.

1 Introduction

Although not always recognized as such, sign languages are full-fledged natural languages that have their own grammar and vocabulary.[1] Contrary to a widespread belief, there is no universal sign language and several hundred sign languages have been documented around the world. The number of deaf people who use them as a primary means of communication worldwide is estimated to be roughly 70 million, and more than half a million in Europe (Way et al., 2024).

[1]This paper is based on Ingimundarson (2026) the first author's master's thesis at the University of Zurich

The field of sign language processing (SLP) is more recent than that of natural language processing (NLP) and has lagged behind, constrained by a lack of technological or computational resources, software, and data (Yin et al., 2021; Müller et al., 2022). This situation has gradually improved but SLP remains underrepresented and fundamental design decisions underexplored (O'Brien et al., 2026). In general, it can also be argued that all sign languages are low-resource languages when compared to spoken languages (Joshi et al., 2020). Nevertheless, larger sign language communities, such as that of American Sign Language (ASL), are comparatively well represented and well resourced, whereas sign languages of smaller communities are low-resource, which restricts the development and deployment of SLP tools based on them.

One such low-resource sign language is Icelandic Sign Language, that will hereafter be abbreviated as ÍTM (***íslenskt táknmál***), which is preferred by the Icelandic Deaf community and ÍTM researchers. ÍTM is now estimated to have about 300 native users, and in total approximately 2,000 users. The total number includes L2 users with late onset of hearing loss, children of deaf adults, foreign L1 signers (ÍTM L2 signers) and hearing signers with various levels of proficiency, the last group comprising somewhere between 1,000 and 1,500 people (Koulidobrova and Sverrisdóttir, 2021).

Automatic processing of ÍTM has not taken place so far and this paper represents one of the first steps in that direction. The task at hand, isolated sign language recognition (ISLR), is not the same as translation, and therefore less practical than a translation system. Nevertheless, for a low-resource sign language such as ÍTM, ISLR is a good starting point. With that in mind, this paper explores how well ISLR can perform in a real low-resource setting, on limited ÍTM data.

The contributions of this paper include:

Proceedings of the Second Workshop on Sign Language Processing, pages 84–95
October 29, 2026 ©2026 Association for Computational Linguistics

1. The first-ever ISLR experiments on ÍTM data
2. Comparison of SPOTER and OpenHands, two open-source frameworks for ISLR
   (a) SPOTER yields significantly better baseline results than OpenHands on all three tasks, outperforming it by 36.36, 14.53 and 6.95 in test accuracy
   (b) Multilingual training in OpenHands improves the test performance by 27.8 and 27.45 percentage points
   (c) Pretraining on ASL data and finetuning on ÍTM data improves SPOTER test performance by 22.73, 23.93 and 14.25 percentage points
   (d) MediaPipe pose estimates in SPOTER performed better than AlphaPose and SDPose pose estimates

2 Related Work

Automatic sign language recognition (SLR) can be split into two tasks, *continuous* or *isolated* SLR (Koller, 2020). In continuous SLR (CSLR), the recognition is performed either directly on the continuous stream or with an intermediate segmentation step, where sign boundaries are identified and the recognition is then performed on isolated signs based on the assumption that a single sign/gloss is contained within the segment boundaries. Isolated SLR (ISLR) is performed on single sign input.

The recognition is then treated as a classification task where information is extracted from signed video input that is processed into a representation suitable for a downstream task (Way et al., 2024). The output labels can be, for instance, words or lemmas in the corresponding spoken language or sign language glosses.

2.1 Video-based Recognition

As observed by Way et al. (2024), in the wake of deep learning, the field of SLR has witnessed a surge of new techniques, models and model architectures. One of the first end-to-end approaches to SLR was the work of Camgoz et al. (2017), who drew on speech recognition and the use of Connectionist Temporal Classification (CTC) algorithms. The novel architecture they proposed for sequence-to-sequence learning, called *SubUNets*, consisted of three layers: a CNN layer to extract spatial features from image input, bidirectional LSTM layers to temporally model those spatial features, and a CTC loss layer on top. This was followed by the *Sign Language Recognition Transformer* (Camgoz et al., 2020), a unified model which was trained to jointly learn CSLR and translation.

2.2 Pose-based Recognition

Pose estimation provides a lower-dimensional alternative to raw video. Pose estimators extract human skeletal keypoints, creating signer-invariant representations that abstract over clothing, skin color, gender etc., focusing on the most relevant parts of the signal and ignoring irrelevant RGB features in the video. Despite these abstractions, the poses remain human-interpretable, but are not entirely anonymous (Battisti et al., 2024). Previous work has predominantly used two pose estimators: OpenPose (Cao et al., 2021) and MediaPipe (Lugaresi et al., 2019). Neither was developed specifically for SLP, but given their lightweight nature compared to raw videos, they have become widespread in SLP.

A recent and comprehensive comparison study of pose estimators is the work of O'Brien et al. (2026). They compare eight different pose estimators and evaluate them on sign language translation (SLT). They show that four pose estimators achieve higher translation scores than MediaPipe, and suggest that alternatives to MediaPipe should be more strongly considered for pose-based SLT. Well-performing and interesting alternatives include Sapiens (Khirodkar et al., 2024) and SDPose (Liang et al., 2026) – which are the best-performing estimators, but have significant compute requirements – and AlphaPose (Fang et al., 2023), which had the third-best translation performance and the fastest compute speed.

2.3 Publicly Available ISLR Frameworks

SPOTER The *Sign Pose-based Transformer* (Boháček and Hrúz, 2022),[2] proposed using pose estimates from Apple's Vision API to train a word-level SLR model. Along with a novel normalization scheme, they presented new data augmentation methods, namely four different spatial augmentations and adding Gaussian noise, which gave the best results. In addition, they demonstrated how well the SPOTER model performs on limited training data compared to an I3D model in the same setting. Their code and data availability has enabled follow-up works such as Azad and Rahman (2026) who introduce several modifications to the

[2] https://github.com/maty-bohacek/spoter

original SPOTER, use MediaPipe poses instead of Vision API, adapt the implementation to Bengali Sign Language data, and substantially improve the previous results on an existing ISLR dataset.

Boháček et al. (2022) improves the original SPOTER approach by replacing the Vision API poses with MediaPipe Holistic. The poses were adapted to the landmarks used in the original SPOTER, in addition to performing hyperparameter search over augmentation parameters. By doing so, they greatly improved the performance on WLASL100, achieving state-of-the-art results for it at the time. This code is, however, not available.

SPOTER only implements one model architecture (Transformer) and neither pretraining strategies nor multilingual training setups are supported.

OpenHands Another important contribution in terms of open-sourced code is the OpenHands library by Selvaraj et al. (2022),[3] (see also NC et al. (2022)) which aimed to exploit key low-resource language insights from traditional NLP for the benefit of sign languages. The open-source framework they released is no longer maintained, but includes pose estimates of existing datasets for five sign languages and more than 1,000 hours of Indian Sign Language data for self-supervised training.

OpenHands supports four model variants: two sequence-based models, RNN and Transformer, and two graph-based models, a spatio-temporal graph convolutional network (ST-GCN) and a sign language GCN (SL-GCN). Based on the results reported by Selvaraj et al. (2022), the graph-based models outperformed the sequence-based models on all of the datasets where accuracy was reported, with the SL-GCN performing best overall.

The framework also provides a multilingual training configuration that allows combining data from multiple sign languages. Two vocabulary strategies are available: a unified vocabulary, where the original classes from each dataset are "normalized" into English glosses, or a combined vocabulary of the original vocabulary of each dataset.

In its default configuration, OpenHands supports 11 datasets for 7 sign languages (American, Chinese, Indian, Turkish, Greek, Argentinian and German).[4] However, three of these datasets are not publicly available. For the other datasets, the authors provide MediaPipe poses.

[3] https://github.com/AI4Bharat/OpenHands

[4] See https://openhands.ai4bharat.org/en/latest/instructions/datasets.html for an overview of the supported datasets.

Number of videos	1,845
Total length (hours:minutes:seconds)	1:44:29
Number of classes	849
Number of signers	33

Table 1: Overall statistics of ÍTM SignWiki dataset

# Classes	# Samples	Proportion
732	2	86.22%
95	3	11.19%
18	4	2.12%
2	5	0.24%
1	6	0.12%
1	8	0.12%

Table 2: Samples per class distribution in the ÍTM SignWiki dataset. Most classes represented by two samples

3 Dataset

We use the *ÍTM SignWiki* dataset presented in Ingimundarson (2026). It is based on the dictionary component of the Icelandic part of *SignWiki*,[5] a web and mobile platform for sign languages and deaf education. The dictionary is bilingual, where ÍTM sign language videos are mapped to Icelandic words or phrases, and currently contains roughly 13,000 signs and phrases. Although the dictionary is open for public input, the vast majority of the recordings originate from the Communication Centre for the Deaf and Hard of Hearing in Iceland (SHH), contributed by people who have been or are currently employed there, and the recordings date from the early 1990s and up until the present day.

3.1 Dataset Profile

An overview of the dataset is given in Tables 1 and 2. As the numbers illustrate, the dataset is very limited in terms of size. Instead of having many examples on a limited set of classes, there are limited examples for many classes. This is in stark contrast to most ISLR datasets, such as the subsets of the MS-ASL dataset (Joze and Koller, 2019), that range from 100–1,000 classes and have, on average, 25.5–57.4 samples per class (189–222 signers).

As far as the signers in the data are concerned, 84.77% are L1 deaf, 10.30% are hearing and 4.93% are L2 deaf. It is also worth noting that 89.7% of the signers are female and 10.3% are male, and all are white, which gives an indication of the overall composition of the data. As the data has been collected over many years, some of the more fre-

[5] https://is.signwiki.org/

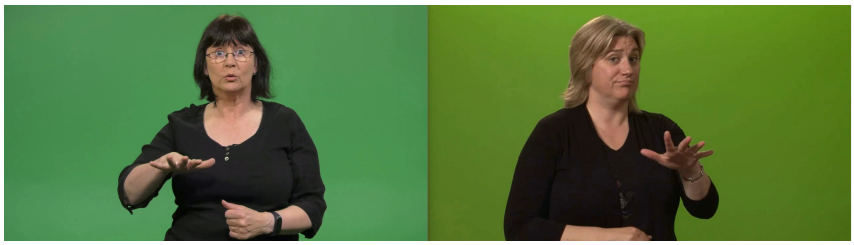


Figure 1: Example of left-/right-handed contrast in the ÍTM SignWiki dataset

quently featured signers have several different appearances, i.e. wearing different clothing, with different hairstyles etc. This would increase the diversity in a video-/RGB-based approach, but is less relevant for pose-based methods.

A further characteristic of the dataset is the fact that the second-most frequently occurring signer in the dataset (Signer 1) is left-handed, the only one out of the 33 signers. The signer appears in 267 out of 1,845 samples. Therefore, many of the classes have a left-/right-hand contrast in addition to the signer difference between training and test samples. In total, 1,075 instances are labeled as right-handed, 202 left-handed, and 568 samples are two-handed symmetrical signs. An example of left-/right-handed contrast is shown in Figure 1.

It is worth noting that a few instances of phrases, as opposed to isolated signs, are included in the data, e.g. *hvað heitir þú* ('what is your name'). The dataset also contains multiple instances of other classes that consist of two or even three signs, such as the word *sérkennari* 'special needs teacher', composed of the signs SÉR ('special'), KENNA ('teach'), and PERSÓNA ('person'). Furthermore, seven of the 849 classes are fingerspelled. It must also be noted that the vocabulary of the dataset is unsystematic and contains, for example, signs for the words *Algeria*, *browned potatoes*, *Lord* (in Christian sense), *feather* and *hippopotamus*.

3.2 Data splits and recognition tasks

Data splits We split the full dataset into a training, validation and test set with 917, 79 and 849 samples, respectively. The difference in size between the validation and test splits is due to dataset constraints. For 38 of the 117 classes that have three samples or more, two samples were retained for training instead of having a validation sample. In many cases, the validation sample is a different recording by the same signer as the training sample, whereas the test sample is drawn from a different signer not represented for that class. Therefore, the performance on the validation set might be misleading with regard to test set performance. This is preferable to validating on an unseen signer while testing on a seen signer for the same class.

Task	Split	# Classes	# Samples
Minimal	Train	22	52
	Validation	22	22
	Test	22	22
Trimmed	Train	117	185
	Validation	79	79
	Test	117	117
Full	Train	849	917
	Validation	79	79
	Test	849	849

Table 3: Recognition tasks based on the ÍTM SignWiki dataset, varying number of classes in the training, validation and test sets

Tasks We define three different recognition tasks with increasing difficulty (varying random chance of success and class balance of training and validation data), see Table 3. The *Minimal* task entails recognizing 22 classes (with ≥ 4 samples per class), the *Trimmed* task has 117 classes (with ≥ 3 samples per class) and the *Full* task has 849 classes (with ≥ 2 samples per class). Moving to the next task means adding more, and more challenging, classes, with the Full task effectively being one-shot recognition over 849 classes.

4 Experiments

We train a series of baselines with OpenHands and SPOTER using only the ÍTM SignWiki dataset (Section 4.1). Then we perform additional experiments on multilingual training (Section 4.2) and a pretraining/finetuning scheme (Section 4.3) where the training data includes other languages as well.

4.1 Baselines

OpenHands We train systems for all four model variants that OpenHands supports: two sequence-

based models, RNN and Transformer (BERT), and two graph-based models, a spatio-temporal graph convolutional network (ST-GCN) and a sign language GCN (SL-GCN) (see Section 2.3).

The framework provides precomputed poses for eight ISLR datasets (listed in Table 8), but also a pipeline to extract MediaPipe poses from video data. We used this pipeline on the ÍTM data. In the native OpenHands setup, the authors then define two different keypoint presets for the pose estimates: *Minimal* (27 2D keypoints of the upper-body, hands and face) or *Top body* (59 keypoints with better coverage of the upper body). Following Selvaraj et al. (2022)'s example scripts we only used the minimal preset in all of our experiments.

In addition to trying different model architectures, we evaluate on all three recognition tasks (see Section 3.2) and either disable or enable data augmentation. Taken together, we train $4 \times 3 \times 2 = 24$ baseline OpenHands models.

In the native OpenHands setup, the validation accuracy is monitored for early stopping, and we followed this example (max epochs=500/1000, patience=80 mode=max). Further hyperparameters were the following: CosineAnnealingLR scheduler with Adam, lr = $1e-3$, and batch size 4/8/16 for *Minimal*/*Trimmed*/*Full*. Models were trained on an A100 or H100 GPU (this is true for all models in our experiments).

SPOTER The best results with SPOTER were achieved with MediaPipe poses as the input representation (see Section 2.3). For this paper, we adapted the SPOTER code to the binary pose format developed by Moryossef et al. (2021a). Inspired by O'Brien et al. (2026) we compare three different pose estimators: AlphaPose (Fang et al., 2023), MediaPipe (Lugaresi et al., 2019) and SDPose (Liang et al., 2026), instead of assuming MediaPipe as a fixture of the experiment.

As described in Section 2, MediaPipe and OpenPose have been the two predominant pose estimators used in SLP in recent years. AlphaPose and SDPose are lesser known in an SLP context. In O'Brien et al. (2026)'s comparison of pose estimators, evaluated on SLT, AlphaPose ranked third, and was found to be more computationally efficient than the better-performing estimators. It is primarily the low-latency benefit of AlphaPose that makes it appealing for the purposes of this project. While working with a low-resource language does not necessarily entail limited computational resources, a lightweight solution with competitive performance would be advantageous.

To extract the poses, we used the *video-to-pose* repository by O'Brien et al. (2026),[6] which at the time of writing supports eight pose estimators in total. In the SPOTER implementation, a total of 54 body landmarks are extracted, including five head landmarks (eyes, ears, and nose) and 21 body landmarks that represent body joints. This results in 54 2D points and an 108 dimensional feature vector for each frame. Following Boháček and Hrúz (2022) we train for 350 epochs and add Gaussian noise to the training set (along with other data augmentation techniques the framework provides).

To summarize, we trained SPOTER baselines for all three recognition tasks (see Section 3.2) and for three different pose estimators ($3 \times 3 = 9$ baseline SPOTER models).

4.2 Multilingual training

We train additional OpenHands models on more datasets in other sign languages (see Table 8 in Appendix A for the full list of datasets).

Vocabulary strategies We experiment with either keeping the vocabularies of all datasets separate (*original*) or unifying them into a single, normalized vocabulary (*unified*). In the first setting, the language code (ISO) of each sign language included is prepended to the class label as a one-hot encoded vector, i.e. *ice__* for ÍTM.

In the second setting, the vocabulary of each non-English/non-ASL dataset is normalized to English glosses. Here, the ÍTM subset has only 817 classes as the normalization allows to combine different sign variants of the same sign, of which there are several examples in the dataset, with the same normalized gloss. See Appendix A for a more detailed explanation.

This results in a single normalized vocabulary and predictions are therefore made with normalized glosses, which can then be mapped back to the original vocabulary, but not to separate variants. As the unified vocabulary experiment effectively is a different version of the *Full* task, we keep those results separate and present them in Appendix A.

These experiments use the multilingual training feature of OpenHands, described in NC et al. (2022), but only the unified vocabulary, but the experiments were inspired by example configs

[6] https://github.com/ZurichNLP/video-to-pose

for multilingual training provided with the framework.[7] We used the following hyperparameters: CosineAnnealingLR scheduler with Adam, learning rate $1e-3$, batch size 64, max epochs 200, with early stopping on validation accuracy (max, patience 30).

Due to time and resource constraints, we evaluate these multilingual models only on the *Full* recognition task.

4.3 Pretraining/finetuning scheme

SPOTER does not support pretraining out-of-the-box. For this paper, we adapted the framework to support pretraining (either with the encoder frozen for a specified number of epochs or unfrozen from the start) and finetuning.

For this experiment, a SPOTER model was pretrained on the ASL Citizen dataset (Desai et al., 2023). It is a community-sourced dataset for ASL that has nearly 84,000 videos filmed by 52 signers and 2,731 classes. One motivation for using it was that MediaPipe poses in pose format for the dataset were already available, though any comparable ISLR dataset for a higher-resource sign language would have been a viable alternative. As MediaPipe outperformed the other two pose estimators in the baseline experiments (see Section 5.1), this setup was exclusively tested with MediaPipe poses. As the aim here was simply for the encoder to learn general representations of ASL and not to achieve the best results on the dataset, the training was limited to 30 epochs (test accuracy 40.28).

For the finetuning we tested two approaches: one where the encoder was frozen for the first 30 epochs and only the decoder and head trained, and another where the full model was finetuned from the beginning.

Thus the model is pre-trained on ASL data and finetuned on ÍTM, with the encoder either frozen from the start or unfrozen; exclusively with MediaPipe poses (3×2 experiments).

4.4 Evaluation Metrics

We report test accuracy and validation accuracy, given that the validation set is somewhat particular and limited in two ways (see Section 3.2): on the one hand it covers only 79 classes (compared to the total of 849) and on the other hand, the validation samples are mostly with the same signer as (one of) the training sample(s), whereas the signer in the test set is always unseen for that particular sign.

[7] See https://github.com/AI4Bharat/OpenHands/tree/main/-examples/configs/multilingual

Task	Model	Augmentation	Validation	Test
Minimal	LSTM	✓	18.20	9.09
		-	9.1	0.00
	Transformer	✓	22.70	9.09
		-	31.8	9.09
	ST-GCN	✓	13.60	9.09
		-	40.9	**13.64**
	SL-GCN	✓	9.10	4.54
		-	54.5	9.09
Trimmed	LSTM	✓	3.80	0.00
		-	2.5	0.85
	Transformer	✓	3.80	1.70
		-	7.6	2.56
	ST-GCN	✓	21.5	2.56
		-	51.9	3.42
	SL-GCN	✓	10.10	4.27
		-	59.5	**9.40**
Full	LSTM	✓	7.60	0.00
		-	10.1	0.23
	Transformer	✓	2.50	0.00
		-	3.8	0.11
	ST-GCN	✓	22.8	0.71
		-	26.6	1.06
	SL-GCN	✓	17.70	**1.41**
		-	30.4	0.82

Table 4: Accuracy of OpenHands baseline models trained on ÍTM data only (✓=with augmentation, -=only normalization)

5 Results

This section presents the initial ISLR experiments we performed with the two frameworks. An overview of the best results is shown in Table 7 and an additional table for the multilingual results is included in Appendix A.

5.1 Baselines (ÍTM Data Only)

OpenHands Table 4 shows the performance of all OpenHands baselines. The graph-based models generally outperform the LSTM and Transformer models, with a GCN variant achieving the best test accuracy of 13.64% / 9.40% / 1.41% on the *Minimal* / *Trimmed* / *Full* tasks respectively. Only one configuration yielded more than 10% accuracy on the *Minimal* task; an ST-GCN model without any data augmentation. For reference, the best OpenHands baseline model is repeated in Table 7.

Across the baselines, disabling augmentation yields slightly better test accuracy in 8 of 12 pairs and in general higher validation accuracy. However, since the test sets are rather small (22, 117 and 849 examples) these are in fact minor differences.

Task	Poses	Validation	Test
Minimal	MediaPipe	72.72	**50.00**
	AlphaPose	63.63	45.45
	SDPose	63.63	31.82
Trimmed	MediaPipe	72.15	**23.93**
	AlphaPose	59.49	18.80
	SDPose	59.49	17.09
Full	MediaPipe	67.09	**8.36**
	AlphaPose	53.16	3.06
	SDPose	55.70	4.00

Table 5: Accuracy of SPOTER baseline models trained on ÍTM data only, varying the pose estimation system

SPOTER Table 5 shows the performance of all SPOTER baselines. MediaPipe poses consistently outperforms other estimators, for example the test accuracy of MediaPipe is roughly 5 percentage points higher than AlphaPose on all three tasks. In general, the SPOTER baselines show considerably higher accuracy on the test set than comparable OpenHands baselines (see above). For reference, the best SPOTER baseline model (only MediaPipe results) is repeated in Table 7.

5.2 Multilingual Training

Multilingual training results with the original vocabulary are shown in Table 7, for a direct comparison with baseline scores. Individual per-dataset scores are reported in Table 8 in Appendix A. Multilingual training results only concern the *Full* task. Adding multilingual training outperforms the best OpenHands and SPOTER baselines. For example, the test accuracy of the best SPOTER baseline is 8.36, while the test accuracy of the best multilingual model is 28.86. Unifying the multilingual vocabulary (as opposed to keeping separate vocabularies) yields higher accuracy, 29.21, but on a slightly smaller vocabulary (see Section A.1).

5.3 Pretraining / finetuning scheme

Pretraining / finetuning results are shown in full in Table 6. Pretraining on ASL data also outperforms the best baselines by at least 10 percentage points in test accuracy. For instance, on the *Full* task, the best SPOTER baseline achieves 8.36 test accuracy, while the best finetuned model achieves 22.61 accuracy. Furthermore the results demonstrate that freezing the pre-trained encoder at the beginning of finetuning increases the test accuracy by at least 10 percentage points.

Task	Setting	Val	Test	ΔBaseline
Minimal	Frozen	95.45	**72.73**	+22.73
	Unfrozen	95.45	59.09	+9.09
Trimmed	Frozen	82.27	**47.86**	+23.93
	Unfrozen	67.09	26.50	+2.57
Full	Frozen	78.48	**22.61**	+14.25
	Unfrozen	68.35	13.43	+5.07

Table 6: Test accuracy of a SPOTER model pre-trained on ASL and finetuned on ÍTM compared to the baseline in Table 5 (Frozen=Pre-trained encoder frozen for first 30 epochs, Unfrozen=Full model finetuned from start)

6 Discussion

Baselines: SPOTER vs. OpenHands When only using ÍTM data, SPOTER clearly outperforms Openhands on all three tasks: 50.00 vs. 13.64, 23.93 vs. 9.40 and 8.36 vs. 1.41 test accuracy (see Section 5.1). These margins should be read with the size of the test sets in mind. On the *Minimal* task each prediction is worth $1/22 = 4.54$ percentage points, and the gap amounts to 11 vs. 3 correct predictions; on *Trimmed* it is 28 vs. 11 out of 117. The *Full* task, with 71 vs. 12 correct predictions out of 849, therefore carries most of the evidence, and the smaller tasks should not be over-interpreted.

We emphasize that this is a comparison of two frameworks as they are distributed, not of two architectures. Input representation, normalization, augmentation, optimization and checkpoint selection all differ at the same time, and our experiments do not isolate these factors. We therefore offer hypotheses rather than explanations.

For example, OpenHands and SPOTER reduce the full set of MediaPipe keypoints in different ways. OpenHands offers 27-point and 59-point presets, but our OpenHands models use only the 27 point preset, following Selvaraj et al. (2022). SPOTER, on the other hand, uses 54 2D keypoints. This difference in keypoint resolution may in part explain the difference in performance between the frameworks: a coarser hand representation could plausibly mean a disadvantage for OpenHands. Rerunning OpenHands with the 59-point preset would test this directly.

Second, the frameworks normalize differently. OpenHands applies a single shoulder-referenced centering and scaling to the whole skeleton, so hand landmarks occupy a small region of the normalized space, whereas SPOTER normalizes body and hands separately, distorting hand keypoints to

Task	Random	Framework	Model	Setting	Val	Test	Time
Minimal	4.54	OpenHands	ST-GCN	Baseline (No Augmentation)	40.9	13.64	00:01:54
		SPOTER	Transformer	Baseline (MP)	72.72	50.00	00:02:18
		SPOTER	Transformer	ASL-Finetuned (MP)-Frozen	95.45	**72.73**	*05:22:24
Trimmed	0.85	OpenHands	SL-GCN	Baseline (No Augmentation)	59.5	9.40	00:07:39
		SPOTER	Transformer	Baseline (MP)	72.15	23.93	00:18:06
		SPOTER	Transformer	ASL-Finetuned (MP)-Frozen	82.27	**47.86**	*05:25:13
Full	0.12	OpenHands	SL-GCN	Baseline (Augmentation)	17.70	1.41	01:14:32
		OpenHands	SL-GCN	Multilingual Original	68.2	**28.86**	08:15:16
		SPOTER	Transformer	Baseline (MP)	67.09	8.36	01:31:25
		SPOTER	Transformer	ASL-Finetuned (MP)-Frozen	78.48	22.61	*06:07:20

Table 7: Validation and test accuracy on the three ÍTM recognition tasks for OpenHands and SPOTER. (Baseline=best baseline score, Random=Random classification accuracy, MP=MediaPipe Holistic poses, Time=Training time as HH:MM:SS, *=Combined training time (ASL Citizen pretraining took 05:21:11 hours)

a lesser degree. Normalization is intimately tied to generalization; because the test signer is always unseen for a given class (Section 3.2), our test set specifically measures signer-invariant generalization. Consistent with this, SPOTER retains a much larger share of its validation accuracy on the test set (69%/33%/12% across the three tasks) than the best OpenHands models do (33%/16%/8%). Both frameworks overfit to the seen signer; OpenHands does so considerably more. The importance of preprocessing choices of this kind for pose-based SLP has been noted before (Coster et al., 2023; O'Brien et al., 2026).

Third, the training and model selection protocols differ. SPOTER trains for a fixed number of epochs, saves the two checkpoints with highest training and validation accuracy every ten epochs, and evaluates over all of them. Our OpenHands runs use early stopping and checkpoint selection on validation accuracy, which on the *Full* task is computed over 79 of 849 classes, and a single checkpoint was then chosen manually from k saved checkpoints. Model selection is thus both noisier and more weakly related to the target task for OpenHands, and we did not evaluate all saved checkpoints for comparison.

Fourth, and in our view most informative, the graph-based models appear to be data-starved rather than unsuited to the task. The same SL-GCN that reaches 1.41 on the *Full* task with ÍTM data alone reaches 28.86 once other sign languages are added (Section 5.2). Selvaraj et al. (2022) report their graph models performing best on datasets with tens of samples per class; ÍTM SignWiki offers one or two. In this respect our results are in line with Boháček and Hrúz (2022), who show SPOTER learning effectively from small training sets, although their comparison was against a video-based I3D model that must first learn general properties of human motion, whereas both frameworks compared here operate on poses. Our results are therefore consistent with their claim but do not test it under the same conditions.

Finally, although both frameworks use MediaPipe, the poses were extracted with different pipelines: the extraction script shipped with OpenHands in one case and *video-to-pose* (O'Brien et al., 2026) in the other. Differences in version or configuration cannot be ruled out as a contributing factor.

SPOTER: Choice of pose estimator MediaPipe clearly outperformed AlphaPose and SDPose. It is worth reiterating that this was not a comparison of the full pose estimates but of the customized (reduced) SPOTER format. However, the format was the same for all three estimators and the training schemes identical. As mentioned in Section 2, a comparison study of pose estimators performed by O'Brien et al. (2026) revealed that three other pose estimators performed better than MediaPipe on SLT. In our experiments, the downstream task is different, recognition instead of translation, and the set of keypoints is reduced. Nevertheless, MediaPipe appears better suited to ISLR. This is in line with findings of the comparative studies of Coster et al. (2023) and Moryossef et al. (2021b) of pose estimators for SLR, where MediaPipe outperformed OpenPose and MMPose. Our results are, to the best of our knowledge, the first comparison of MediaPipe, AlphaPose and SDPose for SLR.

Cross-lingual transfer effects Both multilingual training and fine-tuning a pretrained model improved the performance, meaning that both con-

stitute genuine cross-lingual transfer effects from higher-resourced sign languages to a low-resource one. Multilingual training raises the best OpenHands result on the *Full* task from 1.41 to 28.86, a twentyfold increase, while ASL pretraining raises the best SPOTER result from 8.36 to 22.61, and by 14–24 percentage points across the three tasks. On the *Full* task the multilingual model is ahead (245 vs. 192 correct predictions of 849).

These two numbers should not be read as a ranking of the two strategies. They come from different frameworks, whose baselines already differ by a factor of six; from different source data, eight datasets in six sign languages in one case and a single ASL dataset in the other; and from different label spaces at inference. The multilingual model predicts over all 5,732 classes of the concatenated dataset, and 137 of its 849 test predictions fall on classes belonging to other sign languages (see Appendix B); these are incorrect by construction, so its ÍTM accuracy is measured under a handicap that the pretrained model does not face. A controlled comparison would require training SPOTER jointly on ASL Citizen and ÍTM and evaluating both schemes within one framework, and the multilingual configuration on the two smaller tasks, neither of which we were able to do.

7 Conclusion

In this paper we explore the performance of two publicly available SLR frameworks in a low-resource setting, using the first ISLR dataset for ÍTM. The dataset is very limited with regard to the intended use in a machine learning task; it has many classes (849) and few samples per class, only two samples for the majority of classes (732). The experiments therefore explored how well SLR can perform in a real low-resource setting.

We demonstrate that multilingual learning and cross-lingual transfer can benefit lower-resource sign languages. In the OpenHands experiments, the multilingual training scheme achieved the best results on the full dataset. In the case of SPOTER, we show that SPOTER learns effectively from very small training sets, corroborating the findings of Boháček and Hrúz (2022). Additionally, finetuning on ÍTM after pretraining on ASL Citizen data yielded the best overall results on the original ÍTM vocabulary on two out of three tasks and demonstrates the effectiveness of a strong pre-trained checkpoint that can be finetuned to different tasks. The multilingual model with original vocabulary in OpenHands yields the best performance on the full (most challenging) task.

The repositories of the adapted versions of the two frameworks have been made publicly accessible on GitHub, in the hope that they may prove useful to other researchers,[8,9] and the dataset will soon be published as well.

8 Limitations

Framework choice Both of these frameworks are from 2022 and comparison with at least one newer ISLR method would have been preferable, but no more recent, publicly available code could be found, except extensions of SPOTER.

Error analysis Beyond general accuracy measures, we present no error analysis. Given the composition of the dataset, it could for instance be insightful to analyze errors based on sign types, e.g. one-handed vs. two-handed, fingerspelled signs and multi-sign compounds (see Section 3.1), or per-signer performance.

Baseline tuning The discrepancy in performance in the OpenHands baselines with regard to data augmentation, where the performance without augmentation was in general better, was unexpected and would need further inspection. The augmentation techniques used followed the examples of the OpenHands authors, but could perhaps be adjusted better to the ÍTM dataset.

Variability We only present one single run for each training configuration. Training several models with different random seeds would make our results and conclusions drawn from them more robust.

Applicability Although the experiments yielded meaningful results it should nonetheless be stressed that the results do not yet constitute practically applicable performance. ISLR systems can be applied to tasks such as looking for signs in videos or dictionaries, but none of the models we trained could be immediately deployed in such a setting. This would require substantially greater amounts of training data as well as a larger vocabulary.

[8] https://github.com/ZurichNLP/openhands-itm
[9] https://github.com/ZurichNLP/spoter-pose

9 Acknowledgements

We would like to thank three anonymous reviewers for useful comments. MM received funding from the SIGMA project (grant no. G-95017-01-07), supported by the Digital Society Initiative (DSI) at the University of Zurich.

The language of the abstract and discussion sections was refined with a Claude agent.

References

Sayad Ibna Azad and Md. Atiqur Rahman. 2026. *BdSL-SPOTER: A Transformer-Based Framework for Bengali Sign Language Recognition with Cultural Adaptation*, page 304–315. Springer Nature Switzerland.

Alessia Battisti, Emma van den Bold, Anne Göhring, Franz Holzknecht, and Sarah Ebling. 2024. Person Identification from Pose Estimates in Sign Language. In *Proceedings of the LREC-COLING 2024 11th Workshop on the Representation and Processing of Sign Languages: Evaluation of Sign Language Resources*, pages 13–25, Torino, Italia. ELRA and ICCL.

Matyáš Boháček, Zhuo Cao, and Marek Hrúz. 2022. Combining Efficient and Precise Sign Language Recognition: Good pose estimation library is all you need. *Preprint*, arXiv:2210.00893.

Matyáš Boháček and Marek Hrúz. 2022. Sign Pose-based Transformer for Word-level Sign Language Recognition. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, pages 182–191.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden. 2017. SubUNets: End-to-End Hand Shape and Continuous Sign Language Recognition. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3075–3084.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation. *Preprint*, arXiv:2003.13830.

Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2021. OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1):172–186.

Mathieu De Coster, Ellen Rushe, Ruth Holmes, Anthony Ventresque, and Joni Dambre. 2023. Towards the extraction of robust sign embeddings for low resource sign language recognition. *Preprint*, arXiv:2306.17558.

Aashaka Desai, Lauren Berger, Fyodor O. Minakov, Vanessa Milan, Chinmay Singh, Kriston Pumphrey, Richard E. Ladner, Hal Daumé, Alex X. Lu, Naomi Caselli, and Danielle Bragg. 2023. ASL Citizen: A Community-Sourced Dataset for Advancing Isolated Sign Language Recognition. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Hao-Shu Fang, Jiefeng Li, Hongyang Tang, Chao Xu, Haoyi Zhu, Yuliang Xiu, Yong-Lu Li, and Cewu Lu. 2023. AlphaPose: Whole-Body Regional Multi-Person Pose Estimation and Tracking in Real-Time. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(6):7157–7173.

Finnur Ágúst Ingimundarson. 2026. Isolated Sign Language Recognition for Icelandic Sign Language (ITM): Experiments in a Low-resource Setting. Master's thesis, University of Zurich.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The State and Fate of Linguistic Diversity and Inclusion in the NLP World. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Hamid Reza Vaezi Joze and Oscar Koller. 2019. MS-ASL: A Large-Scale Data Set and Benchmark for Understanding American Sign Language. *Preprint*, arXiv:1812.01053.

Rawal Khirodkar, Timur Bagautdinov, Julieta Martinez, Su Zhaoen, Austin James, Peter Selednik, Stuart Anderson, and Shunsuke Saito. 2024. Sapiens: Foundation for Human Vision Models. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part IV*, page 206–228, Berlin, Heidelberg. Springer-Verlag.

Oscar Koller. 2020. Quantitative Survey of the State of the Art in Sign Language Recognition. *Preprint*, arXiv:2008.09918.

Elena Koulidobrova and Rannveig Sverrisdóttir. 2021. How to Ensure Bilingualism/Biliteracy in an Indigenous Context: The Case of Icelandic Sign Language. *Languages*, 6(2).

Shuang Liang, Jing He, Chuanmeizhi Wang, Lejun Liao, Guo Zhang, Yingcong Chen, and Yuan Yuan. 2026. SDPose: Exploiting Diffusion Priors for Out-of-Domain and Robust Pose Estimation. *Preprint*, arXiv:2509.24980.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. MediaPipe: A Framework for Building Perception Pipelines. *Preprint*, arXiv:1906.08172.

Amit Moryossef, Mathias Müller, and Rebecka Fahrni. 2021a. pose-format: Library for viewing, augmenting, and handling .pose files. `https://github.com/sign-language-processing/pose`.

Amit Moryossef, Ioannis Tsochantaridis, Joe Dinn, Necati Cihan Camgöz, Richard Bowden, Tao Jiang, Annette Rios, Mathias Müller, and Sarah Ebling. 2021b. Evaluating the Immediate Applicability of Pose Estimation for Sign Language Recognition. *Preprint*, arXiv:2104.10166.

Mathias Müller, Sarah Ebling, Eleftherios Avramidis, Alessia Battisti, Michèle Berger, Richard Bowden, Annelies Braffort, Necati Cihan Camgöz, Cristina España-bonet, Roman Grundkiewicz, Zifan Jiang, Oscar Koller, Amit Moryossef, Regula Perrollaz, Sabine Reinhard, Annette Rios, Dimitar Shterionov, Sandra Sidler-miserez, and Katja Tissi. 2022. Findings of the First WMT Shared Task on Sign Language Translation (WMT-SLT22). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 744–772, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Gokul NC, Manideep Ladi, Sumit Negi, Prem Selvaraj, Pratyush Kumar, and Mitesh M Khapra. 2022. Addressing resource scarcity across sign languages with multilingual pretraining and unified-vocabulary datasets. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Catherine O'Brien, Gerard Sant, and Mathias Müller. 2026. Convenience code for installing and using several pose estimation systems. `https://github.com/ZurichNLP/video-to-pose`.

Catherine O'Brien, Gerard Sant, Mathias Müller, and Sarah Ebling. 2026. Evaluation of Pose Estimation Systems for Sign Language Translation. In *Proceedings of the LREC 2026 12th Workshop on the Representation and Processing of Sign Languages: Language in Motion*, pages 371–386, Palma, Mallorca (Spain). ELRA Language Resources Association (ELRA).

Prem Selvaraj, Gokul Nc, Pratyush Kumar, and Mitesh Khapra. 2022. OpenHands: Making Sign Language Recognition Accessible with Pose-based Pretrained Models across Languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2114–2133, Dublin, Ireland. Association for Computational Linguistics.

Andy Way, Lorraine Leeson, and Dimitar Shterionov. 2024. *Sign Language Machine Translation*. Springer Nature Switzerland.

Kayo Yin, Amit Moryossef, Julie Hochgesang, Yoav Goldberg, and Malihe Alikhani. 2021. Including Signed Languages in Natural Language Processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7347–7360, Online. Association for Computational Linguistics.

A Multilingual Results

A.1 Multilingual Training

In Table 8, we present the results of the test set evaluation for both vocabulary approaches, unified and original, of the multilingual training, with two different architectures, along with a comparison of the results reported by Selvaraj et al. (2022) and NC et al. (2022).

The multilingual training outperforms the best accuracy reported by Selvaraj et al. (2022) and NC et al. (2022) on five datasets, ASLLVD, GSL, INCLUDE, LSA64 and WLASL, and in four cases out of five with the original vocabulary and an SL-GCN. The discrepancy in performance on the RWTH-PHOENIX dataset is notable. This data differs from the other datasets, as the poses have been extracted from low-resolution image files instead of videos that are furthermore from continuous data.

Our setup is not completely comparable to the work of NC et al. (2022), where the unified vocabulary approach is used. We include the ÍTM dataset, whereas they include – in addition to the datasets we use – one Turkish Sign Language and two Chinese Sign Language datasets, that were not available to us. They explore different pretraining and finetuning options: one where the unified vocabulary (referred to as *MultiSign-ISLR dataset* is pretrained on Indian Sign Language data and finetuned individually, another model pretrained on *SignCorpus*, a large pretraining dataset of 4.6K hours of data from 10 different sign languages, and finetuned individually, and lastly pretrained on *SignCorpus* and finetuned jointly on all of the sign languages.

Original vs. unified vocabulary As an alternative to a simple concatenation of all sign classes from all languages, we consider a **unified** vocabulary. As an example from our dataset, two different classes for sign variants of *allt fínt* 'all good' are mapped to the same normalized gloss, ALL_GOOD. Including ÍTM in this experiment is admittedly only synthetic, as the original data is not glossed nor are there English glosses available. To synthesize this, the classes (words) in the ÍTM data were machine-translated to English, manually reviewed, and compared to normalized glosses for the other datasets – with 40.8% overlap between the translated ÍTM glosses and the other normalized glosses (see NC et al. (2022) for their approach to the normalization).

	Original		Unified		OpenHands	
Dataset	**ST-GCN**	**SL-GCN**	**ST-GCN**	**SL-GCN**	Selvaraj et al. (2022)	NC et al. (2022)
ASLLVD	50.92	**55.64**	49.49	55.38	-	50.1
AUTSL	90.89	91.42	90.65	91.47	91.9	**92.1**
GSL	94.94	**96.08**	94.40	95.62	95.4	93.6
INCLUDE	96.07	96.57	94.85	**97.55**	93.5	96.3
LSA64	96.56	**98.75**	96.56	96.87	97.8	97.5
MSASL	62.20	65.89	61.48	66.44	-	**67.4**
RWTH	0.41	1.03	0.20	0.82	-	**48.5**
WLASL	44.27	**47.43**	44.92	47.01	30.6	46.6
ÍTM (*Full*)	20.84	28.86	22.12	**29.21**	-	-

Table 8: Per-dataset test accuracy for two multilingual models, 1) original vocabulary (5,732 classes) and 2) unified vocabulary (4,261 classes), along with the best standalone results (all SL-GCN) reported by Selvaraj et al. (2022)

Our results demonstrate how the unified vocabulary outperforms the original vocabulary and demonstrate the effectiveness of crosslingual transfer. It must, however, be stressed that the unified vocabulary has fewer labels than the original vocabulary since several classes that have variants in the dataset are collapsed into a single class in the unified vocabulary. The evaluation is therefore done on a test set with 817 classes instead of the full 849, as in the original vocabulary. And consequently, the results of the two vocabulary approaches are not fully comparable and the difficulty of the tasks is a confounding variable.

In a similar vein, regarding the unified vocabulary approach, we emphasize that these results are based on synthetic ÍTM glosses that have not been verified by an ÍTM expert. Therefore, the results are only an indication of the benefits of this approach, whereas the results with the original vocabulary provide more direct evidence of the benefits of multilingual training.

B Multilingual Prediction Analysis

If the original vocabulary is preserved in the multilingual OpenHands model, then the recognition is performed over a large multilingual label space, and predictions in other languages are possible. On the full test set there are 137 cases of another sign language being predicted and the overall distribution is shown in Table 9. This distribution is in line with the number of classes in the included datasets, and ASL has the largest part of the vocabulary in the concatenated dataset. It is worth considering whether these foreign predictions are in fact correct for the other sign languages. As an example of this, one might for instance look at the prediction for the ÍTM class *frumskógur* 'jungle', which is the plural form of tree in ASL (ase__tree_pl), or the ASL prediction *boy* for the ÍTM class 'man'. The similarity at the level of the word label or gloss alone suggests that the ASL predictions may be correct. However, this might also simply be a coincidence and there are multiple examples of foreign predictions with no label similarity to the ÍTM class, although the underlying signs may nonetheless be phonologically similar. This would require a closer inspection and comparison of the videos.

ISO	**SL**	**Count**
ase	American	88
tsm	Turkish	16
gsg	German	1
ins	Indian	20
gss	Greek	9
aed	Argentinian	3
icl	Icelandic	712

Table 9: Multilingual test prediction distribution with original vocabulary (ST-GCN)

Using Sign Phonology to Improve Few-Shot Cross-Lingual Transfer Learning for Sign Language Recognition

Matteo Lanza
King's College School, Wimbledon
matteo.lanza@icloud.com

Abstract

Pose-based sign language recognizers represent the two hands as separate halves of a skeleton graph, so a left-dominant and a right-dominant signer produce the same sign as mirror images, and a model must learn every sign in both orientations. Which hand leads, however, is a fact about the signer, not the sign. We therefore canonicalize dominance, mapping the hands onto sign phonology's dominant/non-dominant roles: any training clip whose left hand carries clearly more motion energy is mirrored – a lossless operation, since the mirror is an exact automorphism of the skeleton. Canonicalized donors trained on Turkish Sign Language transfer to six other sign languages +6.01 accuracy percentage points better than standard donors under full fine-tuning, and +5.19 with frozen features; every canonicalized donor outscores every standard one (exact p = 1/70 over n = 8 donors), the gain is positive on all six recipients, and most of the gain remains even when recipient data is not canonicalized. To locate the source of the gain, we train nine donor-training variants that contain identical data and differ only in which clips are mirrored. Every other variant beats the natural donor, and three that keep the natural left/right proportion recover 57 to 79% of the gain (so a global orientation change is not the whole effect); this narrows the mechanism without identifying it. One variant, a random 20% re-mirroring after canonicalization, outscores the operator itself, so it is sufficient rather than optimal.

The motion-energy comparison itself is fragile: across seven corpora, 0.12–65.1% of clips never have a hand detected, so up to 80.7% of apparently one-handed clips are tracking artifacts rather than signing. Restricting the transfer comparison to clips in which both hands were tracked shrinks the fine-tune gain from +5.72 to +4.77 with full separation (exact p = 1/15 using four standard and two canonical donors). We also release relevant per-corpus validity checks.

1 Introduction

Sign language recognition from video commonly runs on pose skeletons: a pose extractor reduces each frame of a clip to a fixed set of body and hand landmarks, and the recognizer is a graph convolutional network over those landmarks, which are joined by edges that follow the body's structure. Locating these landmarks frame by frame is called tracking, and it can fail (a hand the extractor does not find simply produces no coordinates). The two hands of a signer occupy disjoint blocks of 21 nodes each, so a left-dominant and right-dominant signer produce the same sign in mirrored blocks, and a recognizer must learn it twice. The dominant hand is the one that performs one-handed signs and leads in two-handed ones; which hand that is varies by signer instead of by sign (Battison, 1978; Padden and Perlmutter, 1987). Sign phonology specifies hands by role instead of by anatomical side, and canonicalization converts from left and right hand blocks into these roles, so that the fixed 'right-hand block' is always the dominant hand articulator (Sandler and Lillo-Martin, 2006). The asymmetry can be measured concretely: on our donor corpus of 31,610 clips, 226 signs and 37 signers, whether a clip's two hands can be compared at all carries 0.821 bits about the sign, while which way the comparison resolves carries 0.457 bits about the signer and only 0.010 bits about the sign (Section 3). The paper quantifies information in bits; one bit is the information of a single yes/no answer. This paper concerns transfer: a model is first trained on a large corpus of one sign language (the donor), then adapted, with only a few labeled clips per sign, to each of six other sign languages (the recipients).

We intervene with a single lossless preprocessing decision – dominance canonicalization: if a clip's left hand carries more than $1.2\times$ the right hand's motion energy, the clip is mirrored

Proceedings of the Second Workshop on Sign Language Processing, pages 96–110
October 29, 2026 ©2026 Association for Computational Linguistics

(an operation which is an exact automorphism of the skeleton graph, and so does not lose or add any data). The donors are CTR-GCN (Chen et al., 2021), used unmodified, trained on Turkish Sign Language (Mercanoglu Sincan and Keles, 2020); the question is what and how much they transfer to the six recipient languages. Donors trained after canonicalization transfer +6.01 points better than donors trained on the natural corpus under full fine-tune, and +5.19 points better under frozen prototypes. The separation is complete in both regimes: every canonical donor outscores every standard donor. The gain is positive on every recipient language.

The transform also costs little or nothing on the donor's own language (Section 5).

Per-hand motion energy is a relatively fragile statistic: a hand the extractor never detects has energy exactly zero, so a tracking failure forces the motion energy comparison to a deterministic extreme. Across our seven corpora the share of clips in which a hand is not detected at all ranges from 0.12% to 65.1%, which is a property of the extractor-corpus pairing (Bilge et al., 2024). Section 6.2 measures the effect and re-scores the headline result on only clips where both hands were tracked (which shrinks but keeps complete separation under fine-tune).

This paper makes four distinct contributions: a lossless dominance canonicalization, with its cross-language transfer gain across six recipient sign languages measured; an information-theoretic decomposition of the chirality channel, showing that decidability is lexical while flip decision is signer-based; a mechanism study over nine donor-training variants related by the exact automorphism, in which almost any reassignment of chirality recovers part of the effect, narrowing but not identifying the operative variable; and a seven-corpus tracking census with a validity guard, and adoptable reporting recommendations, released as per-corpus checks (Sections 6.2 and 7).

The recognizer is used unmodified (Chen et al., 2021) and mirroring is standard machinery; the measurement is the contribution. The setting of the research is isolated-sign recognition over the pose-based pipelines that feed sign language translation, and the questions that it addresses are those of knowledge transfer across sign languages, and (through the census) of corpus creation and annotation practices.

2 Related Work

The two hands used to articulate signs do not play interchangeable roles in sign language phonology. Battison (1974) described how variably the non-dominant hand participates in signing, and later formulated the Symmetry and Dominance conditions that constrain two-handed signs (Battison, 1978). Padden and Perlmutter (1987) gave the name 'weak drop' to the deletion of the weak hand in connected signing, where a traditionally two-handed sign is realized with one hand. From this literature, how many hands a sign engages is a part of its lexical identity, even though weak drop can make this variable in practice. Further, which hand specifically is dominant is simply a signer-property, and not a property of the sign itself. The corpora addressed in this paper are all citation-form recordings so where one-handed realizations of a sign are detected, they are not claimed to be a post-lexical application of the weak drop rule.

Börstell (2023) estimated hand dominance in a sign language dictionary from five body landmarks, and later extended the approach to conversational corpus data (Börstell, 2026). Dataset documentation also typically offers little help: in a survey of 108 sign language datasets, only ten report signer handedness at all (Ni et al., 2026). Horizontal-flip augmentation (which is typically harmless in vision tasks) costs 1.3 points of accuracy on WLASL yet is neutral on AUTSL (Kvanchiani et al., 2025), while a comparable ablation on WLASL-100 found a small positive contribution (Laines et al., 2023). Section 6.2 offers a mechanism for why such results should disagree across corpora.

Pose estimators' disproportionate failure on hands is well documented. Roh et al. (2024) report per-frame MediaPipe hand-detection rates of 46.56% on WLASL-100 against 78.83% on AUTSL. O'Brien et al. (2026) compare eight pose estimators and find that those which drop hand keypoints are associated with lower downstream translation quality. Bilge et al. (2024) note that their hand detector 'mostly fails' on AUTSL – which is the cleanest corpus out of the seven we measured here (under our different extractor configuration). Throughout this literature, the standard remedy is the same: missing hands are filled with zeros (Roh et al., 2024; Bilge et al., 2024; Kirandevraj R et al., 2025; Reddy and Kamakshi, 2025). To our knowledge, no prior work reports a whole-clip loss rate (the fraction of clips where a hand is never found

at all), and no prior work connects this missingness to any linguistic property of the signs being lost: O'Brien et al. relate it to BLEU. This paper explores both of these gaps.

Cross-lingual transfer for sign language recognition is well established but not fully understood. Holmes et al. (2023) transfer to "extremely low-resource" Irish Sign Language from three donor languages, and found that neither donor size nor the raw count of shared glosses tracks performance; among the explanations that they raise (but do not test) there is a mismatch in the one-handed/two-handed composition of donor and target lexicons. Bilge et al. (2024) reach a compatible null independently: benchmark subsets that share class names with the training language transfer no better than fully disjoint ones. Our result is a direct test of the 'family' of explanation that both point towards – that what transfers is structural rather than lexical. The nearest explorations: Ovodov et al. (2025) pretrains on a large Russian Sign Language corpus and transfers few-shot; Bilge et al. meta-train on ASL and test on German and Turkish Sign Language (transfer direction inverted relative to ours); Kindiroglu et al. (2024) transfer within language; SignCLIP (Jiang et al., 2024) and OpenHands (Selvaraj et al., 2022) provide multilingual pretraining infrastructure. None of these intervenes on the donor's input distribution, and none measures why the transfer works.

3 The Chirality Channel

Before any learning takes place, we summarize each clip with two numbers. For each hand, we sum the absolute frame-to-frame displacement of that hand's 21 landmarks over both coordinates and every frame of the clip; the resulting quantity is called the motion energy of that hand, which we write as e_L and e_R. At the donor training time, a check is performed between those two numbers: if e_L exceeds $1.2 \times e_R$, the clip is mirrored. Mirroring the clip swaps the two hand blocks, swaps the left and right body nodes, and negates the x-coordinates, such that the busier hand of every mirrored clip sits in the 'right-hand' node block. This operation is called dominance canonicalization. The margin is set at 1.2 rather than 1.0 because near-balanced clips have a noisy energy ratio, and the extra margin keeps the rule from flipping on that noise. Throughout the paper, the term 'chirality channel' represents the single binary fact that

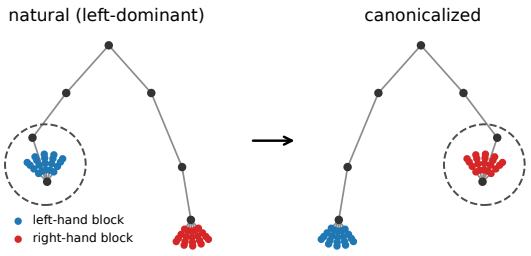


Figure 1: Dominance canonicalization: the mirror swaps the hand blocks and left/right body nodes and negates x, so the busier hand (dashed) always occupies the right-hand block.

this rule manipulates: which node block holds the higher-energy (presumed dominant) hand, and we say that a clip 'flips' when that bit reads left.

Everything that follows is gated by a second derived quantity. The weak-hand share $w = \min(e_L, e_R)/\max(e_L, e_R)$ measures how much of the work the quieter hand does, and we call a clip decidable when w<0.5 – when one hand clearly leads. When both hands work nearly equally, the question 'which hand dominates this clip?' has no answer, and whatever the flip rule decides is arbitrary. This distinction is linguistically grounded. Whether a sign engages one hand or two is part of its lexical identity (Battison, 1978); which physical hand is dominant is a property of the signer (Battison, 1974; Padden and Perlmutter, 1987). Decidability measured at corpus scale is therefore the surface signature of the one-handed/two-handed lexical distinction, including the one-handed surface configurations that weak drop produces.

The mirror operation itself is lossless, as it maps the skeleton's 48 edges exactly onto themselves, and is its own inverse; as a node permutation it fixes only the nose. This makes the mirror permutation an exact automorphism of the graph, so a mirrored clip carries exactly the same information as the original. Two donors trained with the same full 'recipe' on the two mirror-image canonical orientations (all clips left-dominant or right-dominant) transfer within 0.17 accuracy points of each other (range for noise was roughly 1.0 accuracy point). A smaller 6,000 clip in-language probe agrees (0.4 point gap). Therefore, the transform is lossless end-to-end. This means that any effect of canonicalization cannot come from simply mirroring, but instead it must come from which clips

the rule chooses to flip. Losslessness in the flip also means every training variant in Section 6 contains identical information, and differs only in the joint distribution of chirality with signer and sign, so any performance difference is caused by that distribution, instead of a difference in data.

The operator also acts outside of its gate: on the donor corpus, the rule fires on 56.8% of clips, whereas only 46.5% are 'decidable'. Within the decidable clips, the rule behaves as intended, mirroring 88.0% of clips, but it still mirrors 29.7% of undecidable clips. The consequence is that 27.9% of all flips land on clips where there is no 'correct answer' to the question of which hand is participating the most. Clips where neither hand tracked yield $w = 0/0$, and are reported as undecidable (convention effect in Appendix B).

A census reads two binary facts for every clip: whether the clip is decidable (weak-hand share below 0.5, so that a dominant hand exists to be named at all), and the direction the comparison resolves in (meaningful only on decidable clips). We measure how much information each of these carries about the clip's sign and about its signer, as mutual information over the donor corpus of 31,610 clips, 226 signs, and 37 signers. Since MI estimated from finite data is never zero even for unrelated variables, every value is reported beside its shuffle floor (the MI that pure noise produces at this sample size, obtained by shuffling one variable across clips). Decidability turns out to be a fact about the sign: I(decidable; sign) is 0.821 bits, 164 times its floor, while I(decidable; signer) is 0.003 bits, a negligible association sitting at 3.4 times its floor. On the contrary, among decidable clips, I(flip; signer) is 0.457 bits at 256 times its floor, while I(flip; sign) is 0.010 bits, scarcely above its floor at 1.2 times. The research uses this double dissociation: whether the two hands can be compared is a property of the sign, and which way the comparison resolves, where it can legitimately be made, is a property of the signer.

The converse of this claim is also important for the rest of the paper. On the undecidable clips of the same donor corpus, the pattern inverts, with I(flip; sign) rising to 0.285 bits, 30 times its floor, such that the flip decision begins to carry lexical information. This is expected: a hand the extractor does not find has motion energy exactly zero, so tracking failure forces clips into decidability instead of just adding noise. As a result, the recorded flip-direction is related less to the signer's dominant hand than to which hand is lost by the extractor, which depends on the sign being performed. The effect of tracking failure on the decidability of each corpus is reported in Section 6.2.

4 Method

We transfer from Turkish Sign Language into six recipient languages, drawing on seven isolated-sign corpora in which one clip records one sign in citation form. AUTSL (Mercanoglu Sincan and Keles, 2020) is the donor, and the recipients are WLASL-300 (Li et al., 2020), INCLUDE (Sridhar et al., 2020), LSA64 (Ronchetti et al., 2016), Slovo (Kapitanov et al., 2023), BdSLW60 (Rubaiyeat et al., 2025), and SSL400 (Abhishek and Sumanathilaka, 2025; Abhishek, 2024). Every corpus is reduced to one common representation of 49 landmarks per frame, comprising the nose, both shoulders, both elbows, both wrists, and 21 points per hand, all read from MediaPipe Holistic (Grishchenko and Bazarevsky, 2020; Lugaresi et al., 2019). Four of the resulting pose packs come from the OpenHands project's published release (Selvaraj et al., 2022), and the other three we extracted ourselves at version 0.10.21 under a matched configuration, so two separate extraction pipelines produced the pose data used in this paper. What we stage is not always the full published corpus. Table 5 reports each corpus' published scale as well as exactly what we used from it. The donor pack, the largest deviation, comprises official AUTSL train and test partitions minus 274 training clips (0.9%) lost in extraction; the absent validation signers are unlabeled in the OpenHands source release, so no clip was dropped due to tracking quality. The INCLUDE pack similarly ships 264 classes (one more than the published 263). Appendix B gives full details. Every staged subset was fixed by a mechanical rule before any transfer measurement. BdSLW60 caps trials at six per (signer, word). Slovo stages a 120-gloss subset keeping every published clip of each gloss (the corpus ships twenty per gloss); the subset file is part of the release. SSL400 retains every gloss with at least eight extracted clips (the minimum for a k-shot support set plus held-out test) which yields 127 of the archive's 384 gloss folders. No subset was selected on tracking quality or on any measured outcome.

Regarding how splits were conducted: on the donor side, evaluation is signer-disjoint (the donor trains on 26,157 clips from 28 AUTSL signers, and

every donor-side measurement is taken on the 9 signers held out from that training). On the recipient side, splits are signer-disjoint wherever signer labels exist, which is true of four out of six recipients (three cleanly, WLASL machine-clustered). WLASL's signer identities are machine-clustered rather than annotated (Li et al., 2020), while INCLUDE and SSL400 release no usable signer labels at all.

The sign recognizer model is CTR-GCN (Chen et al., 2021), taken unmodified from its original implementation and carrying 1.54M parameters in our 49-node configuration. Adaptation is k-shot, with each recipient supplying $k \in \{1, 5, 10\}$ clips per sign as a support set. Every key result is reported under two different regimes. The first regime is full fine-tune, where all donor weights are retrained on the support set for 30 epochs. The second regime is frozen prototypes, where classification is nearest class-mean on frozen donor features (a stricter probe of representation quality). Each episode fixes a recipient, a value of k, and a random seed that draws support and test clips; one episode under one regime yields one cell.

The headline result compares four standard donors, trained on the natural corpus, with four canonical donors trained after dominance canonicalization. Each donor is then evaluated in the same conditions that it was trained on, which means that canonical donors see canonicalized clips from the recipient corpus, and standard donors see raw clips.

Two separate evaluation grids appear in the paper. The 90-cell basis is made up of 6 recipients, 3 k values and 5 episode seeds, and is the evaluation format used for the headline result. The 54-cell basis, which only differs in having 3 instead of 5 episode seeds, is used for the mechanism-variant comparisons in Section 6. Cell level inference may overstate certainty, and instead donor-level means are the more stable quantity (and all inference below uses them).

The 'effect' is the canonical-arm mean accuracy minus the standard-arm mean. The noise floor is the standard deviation across donors trained with the identical recipe and differing only in seed (effects may also be quoted in terms of floor units). Another relevant factor is the occurrence of separation – where each canonical donor outperforms each standard donor. All confidence intervals are two-sample t intervals (two independent groups: standard and canonical donors) on donor-level means, with six degrees of freedom (number of independent values that are free to vary without breaking constraints estimated from the same data).

			FT acc. (%)		gain	
Recipient	pose	split	std.	canon.	FT	proto
BdSLW60	ours	signer	38.33	47.83	+9.50	+9.86
INCLUDE	OH	clip	55.62	60.85	+5.23	+3.95
LSA64	OH	signer	60.38	66.25	+5.88	+6.78
Slovo	ours	signer	33.10	40.36	+7.26	+5.14
SSL400	ours	clip	31.01	35.59	+4.58	+2.93
WLASL-300	OH	clust.	13.80	17.41	+3.62	+2.49
Pooled			38.70	44.72	+6.01	+5.19

Table 1: Per-recipient transfer, balanced 4-vs-4, 90-cell grid; separation holds within every recipient under fine-tune. Gains are computed from unrounded means. Pose: OH = OpenHands release, ours = extracted locally. Split: signer = signer-disjoint, clust. = signer-disjoint on machine-clustered identities, clip = clip-level, no signer labels.

5 Transfer Results

Table 1 summarizes the transfer results. The most relevant figure is that across six recipient languages, three shot-counts ($k \in \{1, 5, 10\}$) and five episode seeds, in the balanced 4-vs-4 donor design, donors trained after canonicalization outperform standard donors by +6.01 percentage accuracy points (supplemented by a 95% confidence interval on the effect of [+4.62, +7.40]). That is roughly 7.5 times the pooled same-recipe donor spread of 0.80. Every canonical donor outscored every standard donor, and the gain from canonicalization is positive on all six recipients. Each donor is evaluated in the recipient frame (canonicalized or not-canonicalized) it was trained in, so this is a full pipeline contrast. Section 6 separates donor-side and recipient side effects.

Per recipient, the gain under full fine-tune (on the six recipients, three shot-counts and five episode seeds) runs from +3.62 points on WLASL (Li et al., 2020) to +9.50 points on BdSLW60 (Rubaiyeat et al., 2025), against baselines of 13.8-60.4% (averaged over k, with accuracy pulled down by the one-shot cells).

Absolute gain is a poor unit for comparing languages, because it largely measures how many classes a corpus has: it correlates with staged class count at Spearman -0.89, while relative gain correlates at +0.03. In relative terms the effect is largest where transfer is hardest: +26.2% on WLASL and +24.8% on BdSLW60, against +9.4% on INCLUDE. We therefore take relative gain as

the more accurate per-language unit. The effect is robust, and does not hinge on any one language or any one donor. It also does not hinge on how a recipient was staged: the gain is positive on the three OpenHands recipients and the three we extracted ourselves, and under signer-disjoint, machine-clustered and clip-only splits alike (Table 1). Removing any single recipient leaves gain between +5.31 and +6.49, and removing any single donor leaves it between +5.59 and +6.40.

The number of pieces of available labeled recipient data changes the size of the gain but never its sign. Under full fine-tune (on the six recipients, three shot-counts and five episode seeds) the gain is +4.37 at $k = 1$, +7.04 at $k = 5$ and +6.63 at $k = 10$. The dip from $k = 5$ to $k = 10$ is -0.41 points, half a floor unit; a plateau rather than drop. The relative gain moves more steadily, shrinking from +23.9% at $k = 1$ to +16.0% and then +12.3%, which suggests the transform is most important where labeled data is scarcest. Nominal k is a ceiling: at $k = 10$, the signer-disjoint split leaves half of the recipients with 7-9 realized support clips per class.

Canonicalization has a small cost in the donor's own language. Canonical donors score -0.77% accuracy on held-out AUTSL signers (Mercanoglu Sincan and Keles, 2020) in the signer-disjoint probe, with a 95% confidence interval of [-2.77, +1.23]. This rules out a cost above 2.8 points, and rules out the idea that canonical donors transfer better simply because they are more accurate overall. A no-donor baseline trained from random initialization on the support set alone reaches 28.10 pooled: above random selection for every recipient, but below the standard donor by 10.6 points on average (Appendix A.5).

The gain does not shrink as the donor corpus grows (Figure 3, Appendix B). These arms use a 54-cell basis, so the full-size point reads +5.94 rather than +6.01 (90-cell). Under full fine-tune the gain is +3.54 at 7.9k donor clips, then +4.49 at 15.8k and +5.94 at the full 26.2k. Transfer accuracy increase is consistent with log-linearity in donor size over the sampled 3.3× range, at 5.82 points per e-fold for the standard arm, and 7.9 for the canonical arm; the gain between the two rises about 2 points per e-fold. Three sizes cannot test the functional form, so no extrapolation is performed, but within the sampled range, the transform substitutes for data (the canonical donor at 15.8k clips already matches the standard donor at the full 26.2k clips in transfer accuracy).

Under frozen prototypes, where the recipient's encoder is never trained and adaptation reduces to nearest-centroid classification over the donor's frozen features, the balanced 4-vs-4 effect is +5.19 on the 90-cell basis, with complete separation across the eight donors, with a confidence interval of [+2.73, +7.65]. This is roughly 3.7 times this regime's noise floor of 1.42. Its k-curve rises and then flattens, matching fine-tune's k-curve plateau.

Two important factors are slightly weaker here than under full fine-tune, and together justify why every claim in this paper about what the donor's representation has learned is predominantly based on fine-tune results. First, the separation the permutation test rests on is thin: the weakest canonical donor clears the strongest standard donor by 1.54 points against a floor of 1.42 where the fine-tune margin is 4.07 points. Second, that separation holds at only four of the six recipients, failing at LSA64 and SSL400, even though every recipient's mean gain is positive, from +2.49 on WLASL to +9.86 on BdSLW60 (where fine-tune separates all 6).

6 Analysis

6.1 Finding the Operative Variable

This subsection compares donor-training variants that all contain the same underlying data and only differ in which clips are mirrored. Nine variants are trained and evaluated on a common ground of six recipients, three shot-counts and three episode seeds (the 54-cell basis as referred to in Section 5). Each variant changes the same corpus. Natural and canonical variants are identical to those in Section 4. Signer-majority fixes every signer's signs to the signer's majority orientation. The signer-blocked variant also fixes one orientation per signer, but chooses assignments so that the orientation identifies the signer as strongly as possible, while matching the natural left-dominant proportion. The gloss-blocked variant does the same over signs instead of signers. Three random variants canonicalize with no $1.2\times$ margin (mirror iff $e_L > e_R$, so the construction sets the flip rate exactly) and re-mirror a random fraction f $\in \{0, 0.200, 0.568\}$; f=0.568 reproduces the natural left-dominant rate, but now uncorrelated with signer and sign. Each variant changes one or both of the natural couplings.

Every constructed variant beats the natural corpus: against the natural donor's 39.01 ± 1.05, the variants achieve between 42.42 and 46.92 (Table 2). Almost any reassignment of chirality recovers part

of the effect, and the three variants carrying substantial coupling of either kind perform the worst. Three of the variants keep the natural left-dominant proportion of 56.8% exactly: gloss-blocked, signer-blocked and random f=0.568. They still beat the natural donor by 3.4 to 4.7 points, which is 57 to 79% of canonicalization's gain (Table 2). The majority orientation therefore cannot be what separates these donors from the natural one; the difference must lie in how chirality is assigned across clips. The variants that also move the majority into the right-hand block score higher again: unconditional 44.58, canonical 44.96 and f=0.20 at 46.92. That further 0.9 to 3.2 points cannot be attributed by this design to a per-clip rule rather than coinciding with a global flip. The Spearman rank correlation favors the correlation between transfer performance and sign coupling, -0.86 (one-sided p = 0.012) against -0.56 (p = 0.103) for signer coupling, over the seven fixed-preprocessing variants (n=7; flip-augmented row removed due to changing couplings over epochs), without definitively settling which is the dominant variable.

A further donor is trained as a control, mirroring every clip unconditionally, which leaves every coupling identical to the natural corpus. It reaches 44.58 ± 1.07, clearly above the standard donor it was meant to match. It is excluded from the aforementioned rank correlations because we found that AUTSL is stored mirrored relative to its documentation (Limitations), so canonicalization flips the majority of clips, exactly as this control does, and so on this donor a global flip cannot be separated from a per-clip rule (the two donors differ by 0.38 points, inside the seed spread of either arm, so this comparison does not distinguish them). The case against a purely global explanation rests on the marginal-preserving variants above instead.

The second pre-registered prediction was that the signer-blocked variant would reproduce the natural donors. If entropy alone defined transfer, then the signer blocked variant should perform similarly to the natural donors (as they have equal marginal entropy with different assignment). Instead, the signer-blocked donor lands 3.6 points above the natural corpus, falsifying the prediction.

One variant outperforms the canonicalization operator: re-mirroring a random 20% of clips after canonicalization reaches 46.92 ± 0.67 against canonical's 44.96 ± 0.46, consistent with random re-mirroring acting as augmentation (Kvanchiani et al., 2025; Laines et al., 2023). Against the four canonical donors, under fine-tune, the f=0.20 donors separate completely (exact p = 1/35 = 0.029 one-sided, 0.057 two-sided, not corrected for the selection of f as 0.20); under frozen prototypes the two are on par (p=0.31). The value f = 0.20 was selected as the best-performing value from a single-seed hyperparameter sweep. Consequently, given also that the random f variants modify the threshold for a flip decision, the results support only the conclusion that enforcing strict consistency in orientation is not necessarily optimal for transfer under the fine-tuning regime.

Variant	I_{sign}	I_{signer}	FT acc. (%)	% gain
natural	0.383	0.150	39.01 ± 1.05	0
gloss-blocked	0.762	0.001	42.42 ± 0.88	57
signer-blocked	0.172	0.436	42.59 ± 0.19	60
random f=0.568	0.166	0.001	43.69 ± 0.51	79
signer-majority	0.030	0.054	44.51 ± 1.63	92
canonical	0.000	0.000	44.96 ± 0.46	100
flip-aug p=0.5	$\approx 0^{\dagger}$	$\approx 0^{\dagger}$	45.31 ± 1.52	106
random f=0.20	0.049	0.001	46.92 ± 0.67	133

Table 2: Donor-training variants sorted by transfer; couplings in bits, measured after each transform; % gain relative to canonical's, 54-cell grid. The natural and canonical rows are the four standard and canonical donors from the headline result; constructed variants use three seeds each. The f=0 arm (one seed, identical to the canonical operator other than the flip threshold) appears in Table 4. †couplings resampled per epoch.

The ninth variant applies the mirror stochastically rather than as fixed preprocessing: each time a clip is sampled during training, it is mirrored with probability 0.5 (standard flip-augmentation). On the 54-cell grid it reaches 45.31 ± 1.52 (45.66 ± 0.87 with canonicalized recipients), on par with canonicalization. In-language, the flip-augmented variant performs much better: flip-augmented donors reach 80.6 ± 1.1 against the standard donors' 75.8 ± 1.5, and the canonical donors' 75.0 ± 0.5. What canonicalization retains is not transfer accuracy, but a deterministic, invertible, per-clip auditable construct, where flip-augmentation only improves robustness to orientation. The apparent conflict with Kvanchiani et al. (2025) is actually not one: their flips act on RGB frame data, with no usage of skeleton-based nodes, whereas the pose-side mirror is an exact automorphism of the graph.

Under frozen prototypes, constructed variants converge (means of 34.27-36.77, indistinguishable).

Applying canonicalization to donor training data, recipient evaluation data, neither, or both (Table 6)

decomposes the +5.94 total into donor-side +4.22 (71%), recipient-side +0.62 (10%) and interaction +1.11 (19%), with prototypes splitting the same way (71/12/17); the two smaller terms differ by less than the noise floor and we do not order them. The interaction is recipient-concentrated: Slovo supplies about three quarters of it, and it is positive exactly where the recipient-side transform acts often (34.5% of Slovo clips, 21.9% of BdSLW60). The recipient-side gain survives with no donor at all: canonicalizing the recipient alone and using a model trained from random initialization on the support set gains +0.96 points (95% CI [0.41, 1.51], 90-cell), and the gain is confined to those same two recipients: +3.36 and +2.30, against -0.13 to +0.33 elsewhere.

Zeroing one hand's nodes shows which block each donor leans on more. Zeroing the highest-energy block, the standard donor loses 0.80 of its base confidence in the raw frame and 0.79 in the canonicalized frame (so it simply follows whichever block holds the action). Under uniform zeroing of a hand-block on held-out signers, the standard donor's top-1 accuracy falls to 12.5% with the left block zeroed and 34.2% with the right block zeroed, reflecting the corpus' 57% left-dominant share, rather than a model property. For the canonical donor, top-1 accuracy reduces on held-out signers from 75.9% to 18.0% when the right-hand block is zeroed, against 36.9% when the left block is zeroed, so it has learned to expect the action in the right-hand block. Thus, both donors depend on one block: the canonical-donor assumes which block this is, whereas the standard donor must find which block this is. This is consistent with the canonical donor exploiting a fixed dominant block. Together, the variants establish that the joint distribution of chirality with signer and sign is what moves transfer accuracy across them. Whether dominance canonicalization itself is the operative mechanism remains open: this study narrows it without identifying it.

6.2 Tracking Loss

A hand the extractor never finds has motion energy of exactly zero, so tracking failure forces the two-hand comparison to a confident answer rather than blurring it (Section 3).

Whole-clip hand loss spans 0.12% to 65.1% across the seven corpora (Table 3). The rate is a property of the detector-corpus pairing. Within BdSLW60, 94.7% of signer-by-word groups lose a hand in all clips or no clips, so loss is consistent within signer-by-word groups; across detectors, the one that we use fails most on glove-recorded LSA64 (Ronchetti et al., 2016) while Bilge et al. (2024) reports theirs failing on AUTSL, the corpus which ours tracks best.

Corpus	pipeline	loss %	dec. %	corr. %
AUTSL	OpenHands	0.12	46.5	0.3
INCLUDE	OpenHands	0.30	42.6	0.7
Slovo	ours	20.25	51.3	38.8
WLASL-300	OpenHands	31.89	45.7	69.5
SSL400	ours	47.84	63.0	69.3
BdSLW60	ours	48.35	63.6	71.2
LSA64	OpenHands	65.06	78.0	80.7

Table 3: The census: whole-clip loss (a hand never detected in any frame), decidable fraction, corruption share; strict 0/0 convention (lenient convention in Appendix B.1).

A lost hand forces a confident one-handed reading, so the decidable pool mixes genuinely one-handed clips with tracking failures. The proportion of that pool made up by tracking failure (called corruption) rises from 0.3% to 80.7% across corpora. The census also shows that the genuine decidable fraction is of the same order in every corpus (0.20 to 0.46, unrelated to loss), so a corpus's corruption overall can be approximated from the loss rate alone. Predicting using the loss rate reproduces every measured share within 2 points (closed form in the appendix).

By re-scoring each corpus using only the clips in which both hands were tracked, we aim to see if tracking loss is corrupting the two-hand comparison. The relevant statistic is the weak-hand share w, which should be a property of the sign, so we measure its gloss-ICC (how consistent it is across different clips of the same sign). Where loss is negligible the filter changes nothing: AUTSL stays at 0.890 and INCLUDE at 0.917. Where loss is high, most of the apparent lexical signal disappears – WLASL falls from 0.687 to 0.366 and BdSLW60 from 0.739 to 0.252. This means (at the surface level) that roughly half to two-thirds of apparent lexical structure in w on the lossiest corpora was tracking artifact. The operator's own decisions carry the same trend in both directions: on BdSLW60, 37.1% of the clips it mirrors are mirrored because the other hand was never found, and on LSA64, 66.9% of the clips it does not mirror could never have been mirrored at all, because the hand that would have won the comparison was

never tracked.

A direct check re-scores the transfer contrast on well-tracked clips only. Under fine-tune the effect moves from +5.72 to +4.77 with full separation retained, and under frozen prototypes from +5.39 to +4.78. This check is on a smaller scale than the headline grid – a 15-cell fine-tune re-run and a 45-cell prototype re-scoring, over five recipients (since SSL400 was staged after the audit) and over four standard against two canonical donors. The shrinkage thus has a lower precision than the headline effect. Some part of the original gain is attributable to tracking failures. However, a clear gain remains when the comparison is restricted to well-tracked clips.

Re-deriving every flip decision from wrist travel alone (Börstell, 2023, 2026), which survives hand-model failure, moves the donor's flip rate from 56.8% to 57.5% (Cohen's κ = 0.82 against the hand-landmark-displacement rule), so the donor-side term is not a tracking artifact, though both statistics are proxies.

7 Conclusion

Mirroring a donor clip so that its higher-energy hand always occupies the same node block is an exact automorphism of the skeleton graph, relabeling hands into sign phonology's dominant and non-dominant roles. Applied to a Turkish Sign Language donor (Mercanoglu Sincan and Keles, 2020), this process adds +6.01 points of k-shot transfer to six sign languages under full fine-tune and +5.19 under frozen prototypes, with full separation in both regimes. It is sufficient rather than optimal: a random 20% re-mirroring of clips after canonicalization outperforms it, and variants carrying substantial chirality-signer or chirality-sign coupling perform worst, which narrows but does not define the operative property. A donor that mirrors every clip unconditionally reaches 44.58 against canonicalization's 44.96, a difference of about 0.38 points (inside the seed spread of either arm). AUTSL is stored mirrored and originally has a very strong right-dominant majority, so canonicalization flips the majority of its clips and the two largely coincide on the donor; separating them requires a donor corpus that is more mixed in dominance. What the variants do establish is that a global orientation change is not the whole effect: three variants that keep the natural left-dominant proportion still recover 57 to 79% of the gain. Companion measurements report whole-clip hand loss spanning from 0.12% to 65.1% and its effect.

Some practices follow from the results and are cheap to adopt. It can be useful to report the decidable fraction beside any statistic that compares two hands, so that a reader can see how much of the comparison 'had a correct answer' at all. Interpolation of a wholly-missing hand can be misleading if treated as repair: reconstructing 21 landmarks from a wrist position can impute the artifact rather than removing it cleanly, and both zero-filling and interpolation are current practice (Roh et al., 2024; Moryossef, 2024).

Limitations

Every conclusion drawn in this paper is based on results from a single backbone architecture – the CTR-GCN (Chen et al., 2021), which is used unmodified. Two separate regimes were employed (full fine-tune and frozen prototypes), but both are fundamentally based on an identical architecture. This means that results are established for this architecture, but not necessarily for others.

The seven pose datasets come from two different extraction pipelines at matched model complexity: four are from the OpenHands release (Selvaraj et al., 2022), and three we extracted ourselves with MediaPipe Holistic 0.10.21 (Grishchenko and Bazarevsky, 2020). The whole-clip loss census is a measurement under those settings, and its sensitivity to other extractor settings is unmeasured. The census measures whole-clip absence only, and not mid-clip dropout. However, when mid-clip dropout occurs, the coordinates of nodes are carried forwards into the next frame unchanged, which allows the hand to pass the check for its presence, meanwhile it also contributes no energy. This type of transition occurs at varying rates, from 9.5% of frames in AUTSL to 40.3% of frames in SSL400. This biases the weak-hand comparison in the same direction as whole-clip loss, meaning that the corruption shares that are described here are lower bounds.

The donor corpus is not necessarily representative of the property that the transform acts on: AUTSL's left-leading share is 57%, against 1.7-34.5% across the six recipients. The transform therefore re-orients most of the donor corpus used in this paper, but would likely re-orient a smaller fraction of clips for a different donor corpus. A check performed with wrist-

travel signer-dominance estimation (Börstell, 2023, 2026), which survives hand-model failure, reads 34/37 AUTSL signers as left-dominant against documentation reporting 41/43 right-dominant (Mercanoglu Sincan and Keles, 2020). Conversely, the same estimation setup estimates WLASL as 87% right-dominant. This strongly suggests that there is a horizontal-mirroring convention in the distribution/extraction of the AUTSL dataset. Separating majority-side convention alignment from per-clip dominance requires a corpus with a more mixed handedness, verified independently of its documentation.

Signer-disjointness is limited by the labels available: INCLUDE and SSL400 release none, so those two splits are clip-only, and WLASL's signer identities are machine-clustered rather than annotated (Li et al., 2020), so that split rests on the clustering's quality. The donor and the remaining three recipients are signer-disjoint.

The mechanism through which canonicalization improves transfer accuracy is narrowed, but not fully understood. The constructed donor-training variants score substantially higher than the natural arm, which indicates that a reassignment of the chirality channel recovers most of the effect without reproducing it. An exploratory rank correlation favors the sign coupling (Spearman -0.86 against -0.56 for the signer coupling), but this is likely insufficient evidence to conclude that sign coupling is definitively the operative variable.

Two baselines still remain untested and are identified as future work. A test-time mirror-and-average baseline, which averages each clip's embedding with that of its mirror before nearest-centroid classification under frozen prototypes, would show whether any of the gain is available without retraining. A gated rule that mirrors only decidable clips would show whether the 27.9% of flips that land on undecidable clips are beneficial for accuracy or not (the current rule fires on 56.8% of clips while only 46.5% are decidable).

The mechanism arms and the 2×2 decomposition use the slightly smaller 54-cell basis (3 seeds instead of 5). This simply means that these numbers are not directly commensurable with the 90-cell headline, and no comparisons are made across them.

Ethics Statement

Tracking Numbers

Every tracking-loss statistic we report describes the interaction of one specific hand-landmark extractor configuration with a corpus' recording conditions and the phonology of its signs, and none of them is a property of a corpus itself. LSA64's signers wore colored gloves in a 2016 acquisition (Ronchetti et al., 2016), so what is measured there is simply how a modern RGB hand detector meets that recording protocol, without commenting on the corpus further.

Provenance and Licenses

The corpora reach us through four provenances. AUTSL, WLASL-300, INCLUDE and LSA64 arrive via the OpenHands pose release (Selvaraj et al., 2022) (CC BY 4.0) and are not redistributed; of these, WLASL is web-scraped with questioned consent status, so only its pose derivatives are used. Slovo is under a custom license (based on, but explicitly not, CC BY-SA 4.0); we use a public subset, extract poses locally, and store no video beyond processing. BdSLW60 (CC BY-NC-ND 4.0) permits no derivative release, so we publish only aggregate statistics from it; SSL400 (CC BY-NC-SA 4.0, Kaggle) likewise has no pose data redistributed. Across all seven corpora we release only split definitions, MIT-licensed code, and aggregate statistics (no pose data).

Signer Identifiability

Pose extraction removes a signer's appearance, but it does not definitively make a signer anonymous (as they can still be identified by skeletal motion). Therefore, we do not perform any biometric identification of any kind, and there is no effort made to characterize individual signers. The signer identities used are the datasets' own anonymized labels, and their only role in this work is to construct evaluation splits in which no signer appears in both train and test.

Positionality

The author is a hearing researcher, and no Deaf researchers or community members were involved in the design of this study. As a result, no claims are made about community priorities or the desirability of recognition technologies and their development. What is reported is a measurement study of isolated-sign recognition on citation-form corpora,

a constrained task that is not equivalent to interpreting continuous natural signing. Its positive results do not imply a usable sign-language understanding system. The intended near-term use is instead narrower: reducing the labeled-data burden for communities/individuals constructing recognition tools for (under-resourced) sign languages. All seven corpora record urban, deaf-community, citation-form signing, and no claim extends past that setting.

References

Yohan Abhishek. 2024. SSL400 — Sri Lankan sign language dataset. Kaggle, version 6. Licence CC BY-NC-SA 4.0.

Yohan Abhishek and T. G. D. K. Sumanathilaka. 2025. SSL400 — a comprehensive word level dataset for Sinhala sign language recognition. In *2025 5th International Conference on Advanced Research in Computing (ICARC)*, pages 1–6, Belihuloya, Sri Lanka. IEEE.

Robbin Battison. 1974. Phonological deletion in American Sign Language. *Sign Language Studies*, 5(1):1–19.

Robbin Battison. 1978. *Lexical Borrowing in American Sign Language*. Linstok Press, Silver Spring, MD.

Yunus Can Bilge, Nazli Ikizler-Cinbis, and Ramazan Gokberk Cinbis. 2024. Cross-lingual few-shot sign language recognition. *Pattern Recognition*, 151:110374.

Carl Börstell. 2023. Extracting sign language articulation from videos with MediaPipe. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, NEALT Proceedings Series No. 52, pages 169–178, Tórshavn, Faroe Islands. University of Tartu Library.

Carl Börstell. 2026. Seeing who is signing and with which hand. In *Proceedings of the LREC2026 12th Workshop on the Representation and Processing of Sign Languages: Language in Motion*, pages 42–50, Palma, Mallorca, Spain. ELRA Language Resources Association (ELRA).

Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. 2021. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13359–13368.

Ivan Grishchenko and Valentin Bazarevsky. 2020. MediaPipe Holistic — simultaneous face, hand and pose prediction, on device. Google Research Blog. Accessed 2026-08-04.

Ruth Holmes, Ellen Rushe, Mathieu De Coster, Maxim Bonnaerens, Shin'ichi Satoh, Akihiro Sugimoto, and Anthony Ventresque. 2023. From scarcity to understanding: Transfer learning for the extremely low resource Irish Sign Language. In *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 2000–2009.

Zifan Jiang, Gerard Sant, Amit Moryossef, Mathias Müller, Rico Sennrich, and Sarah Ebling. 2024. SignCLIP: Connecting text and sign language by contrastive learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 9171–9193, Miami, Florida, USA. Association for Computational Linguistics.

Alexander Kapitanov, Karina Kvanchiani, Alexander Nagaev, and Elizaveta Petrova. 2023. Slovo: Russian sign language dataset. In *Computer Vision Systems — 14th International Conference, ICVS 2023, Proceedings*, volume 14253 of *Lecture Notes in Computer Science*, pages 63–73, Vienna, Austria. Springer Nature Switzerland.

Ahmet Alp Kindiroglu, Ozgur Kara, Ogulcan Ozdemir, and Lale Akarun. 2024. Transfer learning for cross-dataset isolated sign language recognition in under-resourced datasets. In *2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG)*, pages 1–8. ArXiv:2403.14534.

Kirandevraj R, Vinod K. Kurmi, Vinay P. Namboodiri, and C.V. Jawahar. 2025. Continuous fingerspelling dataset for Indian Sign Language. In *Proceedings of the Workshop on Sign Language Processing (WSLP)*, pages 33–38, IIT Bombay, Mumbai, India (Co-located with IJCNLP–AACL 2025). Association for Computational Linguistics.

Karina Kvanchiani, Roman Kraynov, Elizaveta Petrova, Petr Surovcev, Aleksandr Nagaev, and Alexander Kapitanov. 2025. Training strategies for isolated sign language recognition. *Journal of WSCG*, 33(1-2):21–32.

David Laines, Miguel Gonzalez-Mendoza, Gilberto Ochoa-Ruiz, and Gissella Bejarano. 2023. Isolated sign language recognition based on tree structure skeleton images. *Preprint*, arXiv:2304.05403.

Dongxu Li, Cristian Rodriguez Opazo, Xin Yu, and Hongdong Li. 2020. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1448–1458.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. MediaPipe: A framework for building perception pipelines. *Preprint*, arXiv:1906.08172.

Ozge Mercanoglu Sincan and Hacer Yalim Keles. 2020. AUTSL: A large scale multi-modal Turkish sign language dataset and baseline methods. *IEEE Access*, 8:181340–181355.

Amit Moryossef. 2024. Optimizing hand region detection in MediaPipe holistic full-body pose estimation to improve accuracy and avoid downstream errors. *Preprint*, arXiv:2405.03545.

Yiming Ni, Zhi-Qi Cheng, Jiayu Li, and Wei Cheng. 2026. Sign-language datasets at scale: A comprehensive survey on resources, benchmarks, and annotation standards. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL 2026), Main Conference*. ArXiv:2606.19352.

Catherine O'Brien, Gerard Sant, Mathias Müller, and Sarah Ebling. 2026. Evaluation of pose estimation systems for sign language translation. In *Proceedings of the LREC2026 12th Workshop on the Representation and Processing of Sign Languages: Language in Motion*, pages 371–386, Palma, Mallorca, Spain. ELRA Language Resources Association (ELRA). ArXiv:2604.24609.

Ilya Ovodov, Petr Surovtsev, Karina Kvanchiani, Alexander Kapitanov, and Alexander Nagaev. 2025. Logos as a well-tempered pre-train for sign language recognition. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 24340–24353, Suzhou, China. Association for Computational Linguistics.

Carol A. Padden and David M. Perlmutter. 1987. American Sign Language and the architecture of phonological theory. *Natural Language & Linguistic Theory*, 5(3):335–375.

Tatigunta Bhavi Teja Reddy and Vidhya Kamakshi. 2025. Pose-based temporal convolutional networks for isolated Indian Sign Language word recognition. In *Proceedings of the Workshop on Sign Language Processing (WSLP)*, pages 47–50, IIT Bombay, Mumbai, India (Co-located with IJCNLP–AACL 2025). Association for Computational Linguistics.

Kyunggeun Roh, Huije Lee, Eui Jun Hwang, Sukmin Cho, and Jong C. Park. 2024. Preprocessing mediapipe keypoints with keypoint reconstruction and anchors for isolated sign language recognition. In *Proceedings of the LREC-COLING 2024 11th Workshop on the Representation and Processing of Sign Languages: Evaluation of Sign Language Resources*, pages 323–334, Torino, Italia. ELRA and ICCL.

Franco Ronchetti, Facundo Quiroga, César Armando Estrebou, Laura Cristina Lanzarini, and Alejandro Rosete. 2016. LSA64: An Argentinian sign language dataset. In *XXII Congreso Argentino de Ciencias de la Computación (CACIC 2016): Libro de Actas*, pages 794–803, San Luis, Argentina. Red de Universidades con Carreras en Informática (RedUNCI). Also available as arXiv:2310.17429.

Husne Ara Rubaiyeat, Hasan Mahmud, Ahsan Habib, and Md Kamrul Hasan. 2025. BdSLW60: A word-level Bangla sign language dataset. *Multimedia Tools and Applications*, 84(34):42399–42423.

Wendy Sandler and Diane Lillo-Martin. 2006. *Sign Language and Linguistic Universals*. Cambridge University Press, Cambridge, UK.

Prem Selvaraj, Gokul Nc, Pratyush Kumar, and Mitesh Khapra. 2022. OpenHands: Making sign language recognition accessible with pose-based pretrained models across languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2114–2133, Dublin, Ireland. Association for Computational Linguistics.

Advaith Sridhar, Rohith Gandhi Ganesan, Pratyush Kumar, and Mitesh Khapra. 2020. INCLUDE: A large scale dataset for Indian sign language recognition. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20)*, pages 1366–1375, Seattle, WA, USA. ACM.

A Pre-registered Predictions and Outcomes

A.1 The Predictions

The pre-registered predictions are reported with outcomes in Section 6.1: transfer was predicted to fall monotonically as re-mirrored fraction f increased (falsified, further information in A.2), and the signer-blocked variant was predicted to reproduce the natural donors within 1.5 points (falsified cleanly). The third prediction that flip-augmentation would land between canonical and standard is inconclusive, as it achieves accuracy within noise of the canonical donor.

A.2 The Transfer-versus-f Comparison

Testing of one more value of f (0.4) further investigates the effect of f. Note that f=0 is not equivalent to the canonical donor: in the donor variant where a random fraction is mirrored (including f=0) the threshold for an original mirroring is if e_L exceeds e_R, with no $1.2\times$ buffer that is included in the main canonicalization process.

f	n seeds	fine-tune	prototypes
0	1	44.30	34.56
0.20	3	46.92 ± 0.67	36.77 ± 1.77
0.40	1	45.42	35.89
0.568	3	43.69 ± 0.51	35.85 ± 2.53
canonical (ref.)	4	44.96 ± 0.46	36.06 ± 1.54

Table 4: Transfer against re-mirrored fraction f, 54-cell grid. The curve is non-monotone; per the pre-registration it is confounded as a test of orientation entropy and carries no interval or p.

A.3 Why the Sweep Could Not Isolate Entropy

The sweep was designed to price orientation randomness in accuracy points per bit, which presumes that the amount of randomness in orientation is what sets transfer accuracy. However, because an entirely random subset of clips is flipped, orientation carries near zero information about signer and sign, which means that the sweep of f does not actually change the variable which is shown to directly influence transfer accuracy. Regardless, the random-fraction variants remain useful as zero/low coupling variants/baselines in the variant table of Section 6.1.

A.4 Within-Signer Variation

In theory, if each signer's clips all read as one orientation (assuming that they use the same dominant hand for all clips), the chirality-signer coupling would equal the channel's marginal entropy of roughly 0.99 bits: the natural corpus measures 0.150. That means that the gate's per-clip decision is not constant per signer – it instead follows the sign performed and the tracking more than the individual signer. Blocking by signer deletes that within-signer variation, which is a large intervention (coupling goes from 0.150 to 0.436 signer-chirality) despite matching the corpus' marginal statistics (proportion of left-dominant) exactly.

A.5 No-Donor Baseline

This test fine-tunes the same architecture with an empty donor state dict, so the optimizer, schedule and code path are identical to the fine-tuning. Each cell was trained for up to 60 epochs (twice the fine-tune budget), and was scored at whichever level they performed best at (after 30 or 60 epochs). 89 out of 90 cells improved between the two, which suggests that there could still be some room for improvement. Per-recipient accuracy against the chance floor: BdSLW60 28.80 (1.80), INCLUDE 37.63 (0.42), LSA64 52.72 (1.56), Slovo 20.25 (0.83), SSL400 21.15 (0.91), WLASL-300 8.06 (0.35). The deficit against standard-donor transfer grows with support size: -5.81 at k=1, -12.77 at k=5, -13.24 at k=10.

B Full Corpus Tables

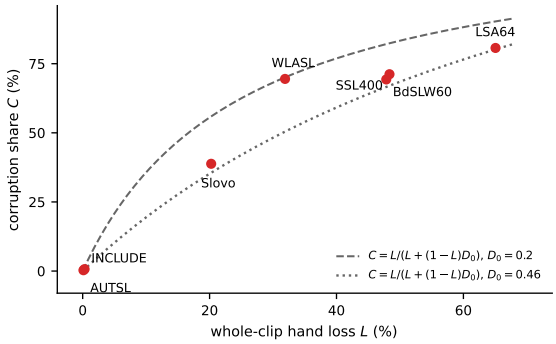


Figure 2: Corruption share versus whole-clip hand loss, with the closed form at the observed extremes of D_0.

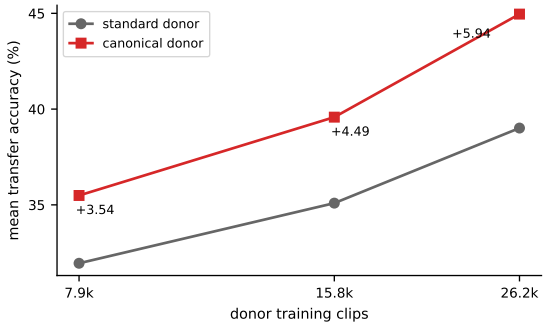


Figure 3: Transfer against donor size, 54-cell grid; only the full size is replicated.

	recipient raw	recipient canon.
standard donor	39.01	39.63
canonical donor	43.23	44.96

Table 6: The 2×2 under full fine-tune, 54-cell grid: total +5.94 = donor +4.22 (71%) + recipient +0.62 (10%) + interaction +1.11 (19%); prototypes 71/12/17.

B.1 The 0/0 Convention

A clip in which neither hand was tracked has weak hand share $w = 0/0$. The main text counts such clips as undecidable ('strict'); counting them as decidable ('lenient') slightly inflates corpora's decidable fractions. Under the lenient convention the decidable fraction rises by 4.4 points on SSL400 (63.0 to

Corpus	Language	Published	Staged pack
AUTSL (donor)	Turkish	226 signs, 38,336 clips, 43 signers	31,610 clips, 37 signers
WLASL-300	American	300 glosses, 5,117 clips, 109 signers	identical
INCLUDE	Indian	263 glosses, 4,287 clips, 7 signers	4,284 clips, 264 classes, no signer labels
LSA64	Argentinian	64 signs, 3,200 clips, 10 signers	identical
Slovo	Russian	1,000 glosses, 20,000 clips, 194 signers	2,400 clips, 120 classes, 106 signers
BdSLW60	Bangla	60 words, 9,307 trials, 18 signers	4,643 clips
SSL400	Sinhala	384 glosses	2,433 clips, 127 classes, no signer labels

Table 5: Published scale versus staged packs. Donor training: 26,157 clips, 28 signers, 9 held out.

67.4%), 3.1 on BdSLW60 and 2.2 on LSA64, and by under 0.5 points elsewhere; the corruption share rises by at most 1.7 points. No conclusion in the paper is dependent on the choice. The release audit also tests presence against a 0.5 frame-fraction threshold, which achieves exactly equivalent results to the never-detected definition on all staged tests. This holds for hands recorded as absent, where per-clip presence is 0 or 1 for every hand-block. This does not address continuous tracking issues – where the extractor drops a hand mid clip the previous coordinates are carried forward, so the hand still reads as present. Loss is therefore recorded as whole-clip granularity only.

B.2 The Corruption Closed Form

Writing L for a corpus' whole-clip loss rate and D0 for the fraction of its cleanly tracked clips that are genuinely decidable, the corrupted share of the decidable pool is C = L / (L+(1-L) $\times$ D0). This is monotone in L by construction, and reproduces every measured share above within 2 points (figure alongside). D0 remains between 0.20 and 0.46 in all seven corpora.

B.3 The Guard-Test ICC Table

Gloss-ICC of the weak-hand share w, scored on all clips, and then under the guard of only when both hands are tracked, with the vertical position channel as the specificity control.

Corpus	w all	w grd.	vert. all	vert. grd.
AUTSL	0.890	0.890	0.707	0.707
INCLUDE	0.917	0.917	0.639	0.638
Slovo	0.693	0.592	0.474	0.540
WLASL-300	0.687	0.366	0.300	0.327
BdSLW60	0.739	0.252	0.498	0.581
LSA64	0.777	0.560	0.769	0.789
SSL400	0.683	0.526	0.323	0.412

Table 7: Gloss-ICC of the weak-hand share w and of the vertical channel, on all clips and under the both-hands-tracked guard. The w column falls in every lossy corpus while the vertical column is flat within 0.001 or rises in all seven.

B.4 Cell-Level Grids

Full cell-level grids ship in the full release as JSON.

C Reproducibility

C.1 Seed-to-Recipe Map

Seeds	Recipe
0–3	standard donors (no transform)
90, 94, 12, 13	dominance canonicalization
91	all-left mirror-image canonical donor
92–93	second donor-subsample pair (7,910 clips – release only)
40–42	per-epoch flip augmentation, p = 0.5
43–45	unconditional flip (100% mirrored, no gate)
50–57	random-fraction sweep
60–63	donor-subsample scaling pairs
70–72	signer-blocked
75–77	gloss-blocked
80–82	signer-majority

Table 8: Seed-to-recipe map. Checkpoints are named by seed alone; the seed identifies the recipe.

C.2 Hardware and training

Every pose is reduced into a 49 node, 2D skeleton over 32 frames: 7 body nodes (nose, both shoulders, both elbows, both wrists) plus 21 landmarks per hand, taken from MediaPipe Holistic. Only the x-y image-plane coordinates are used, which

means that MediaPipe's z channel and the per-point confidences are discarded.

The model used is an unmodified CTR-GCN (1.54M parameters in the 49-node configuration, with a 48-edge skeleton graph). Donor training specifications: 35 epochs, SGD (lr 0.1, Nesterov momentum 0.9, weight decay 4e-4, cross-entropy with label smoothing 0.1), batch 256, 5-epoch linear warmup then $0.1\times$ step decay at 60% and 85% of training. Recipient adaptation specifications (fine-tune regime): 30 epochs, SGD (lr 0.01, momentum 0.9, weight decay 1e-4), batch 128, no scheduling, individual classification head sized to the episode's class count. Frozen-prototype regime: no training at all – nearest class using L2 on normalized donor features.

All training and evaluation ran on single commodity GPUs (RTX 4090 class), or an Apple M4 series Macbook.

C.3 Release Contents

Code (MIT license), split definitions, per-corpus validity checks and all results in JSON are released at `https://github.com/Pwindle1/sign-transfer-learning`. Pose data is not included (Ethics Statement).

Recognizing Uruguayan Sign Language Glosses in Continuous Discourse: A Curated Dataset and First Baselines

Manu Bervejillo[1,*] **and Facundo Techera**[1,*] **and Emilio Valle-Lisboa**[1,*]
Ariel E. Stassi[2] **and Luis Chiruzzo**[1]
[1]Instituto de Computación, Facultad de Ingeniería, Universidad de la República, Uruguay
[2]Departamento de Ingeniería Biológica, Universidad de la República, Uruguay
[1]{manu.bervejillo,facundo.techera,emilio.valle.lisboa,luischir}@fing.edu.uy
[2]astassi@cup.edu.uy

Abstract

Uruguayan Sign Language (LSU) is a low-resource language with a lack of computational models. Building on the first gloss-annotated corpus of LSU interpreter videos, we release LSU-446, a curated dataset of 17,793 sign instances over 446 glosses. Unlike standard benchmarks recorded under controlled conditions, LSU-446 consists of interpreted discourse, with an average sign duration of 403 ms, and heavy coarticulation. To the best of our knowledge, this work establishes the first isolated sign recognition baselines for LSU at scale, comparing a Transformer (SPOTER) and a graph convolutional network (MSG3D); MSG3D reaches 32.7% Top-1 over 446 classes and, unlike SPOTER, degrades only marginally as the vocabulary grows. We then use both architectures for a sign spotting experiment on a held-out video from an unseen signer and domain (33.8% out-of-vocabulary), comparing a segmenter-based pipeline against a sliding window, under an evaluation protocol that decouples temporal detection from lexical recognition.

1 Introduction

Sign languages are full natural languages with their own phonology, grammar and lexicon, generally used by the deaf and hard of hearing communities. There is no universal sign language, as each community develops its own. In Uruguay, roughly 30,000 people are deaf and a further 120,000 are hard of hearing (Ministerio de Salud Pública de Uruguay, 2023), yet computational work on Uruguayan Sign Language (LSU) remains largely unexplored, with limited works like LSU-DS (Stassi et al., 2022), a studio-recorded dataset of 23 isolated signs in which each sign begins and ends at a neutral rest pose; and (Chiruzzo et al., 2026), an experiment on MT between gloss-based LSU and oral Spanish. To our knowledge, no model, benchmark or tool addresses LSU recognition at a realistic vocabulary size or in continuous discourse.

Building annotated data for any sign language is costly: it requires annotators fluent in both the language and its transcription conventions, and there is no equivalent of automatic speech recognition to accelerate the process (Moryossef et al., 2021). What makes LSU low-resource, however, is not merely this lack of prior work, but that its data remains scarce even after annotation effort. The LSU corpus used in this work was obtained from real-time interpreted sign language videos consisting of professional interpreters on television and in narrated stories, rather than clips of isolated signs recorded specifically for this purpose, and is consequently small and lexically imbalanced to begin with: out of 4,612 raw gloss variants, only 446 reach the ten instances we require for training (§3). Its signs last 403 ms on average — against 2.41 s in WLASL (Li et al., 2020a) — are heavily coarticulated, and usually do not reset to a neutral rest pose between lexical items.

We present the first systematic study of automatic LSU recognition at scale. Building on a first gloss-annotated LSU corpus (Fojo et al., 2024) we address two problems: *isolated sign language recognition* (ISLR), which classifies a previously segmented video as a single sign, and *sign spotting*, which locates and labels signs within a continuous stream. Both ISLR and sign spotting differ from continuous sign language recognition (CSLR), which transcribes whole sentences into their corresponding gloss sequences, without the need of per-gloss timestamps. This work does not address CLSR. Since no prior LSU system exists for sign spotting, our aim is a first baseline along with an analysis of different methods and architectures to solve it.

[*]Equal contribution.

Proceedings of the Second Workshop on Sign Language Processing, pages 111–120
October 29, 2026 ©2026 Association for Computational Linguistics

Main contributions of this work

1. **LSU-446**, the first curated dataset for LSU sign recognition (17,793 instances of pose landmarks over 446 glosses), with a reproducible preprocessing pipeline (§3).

2. The **first ISLR baselines for LSU at a realistic vocabulary size**, comparing two state-of-the-art methods, and showing one model to be remarkably more robust as the vocabulary grows (§4).

3. A **three-level evaluation protocol** for sign spotting (unlabeled temporal detection, Top-5 and Top-1 labeled lexical recognition) reporting F_1 and Intersection-over-Union (§5.2).

4. An **empirical comparison** of segmenter-based and sliding-window spotting approaches on a video with an unseen signer and an unseen domain (§5.6).

2 Related Work

Gloss representation. Sign language corpora are commonly annotated with *glosses*: labels written in the main oral language spoken in the region (in our case Spanish), that identify the sign being produced at each point in a video (Fojo et al., 2024). A gloss is not a translation — sign languages have their own grammar and lexicon, so the correspondence between a sign and its gloss is a simplification adopted for annotation and computational processing. The representation is also lossy, discarding non-manual and prosodic information, and its granularity depends on the annotator convention rather than a fixed standard (Zhang and Duh, 2021). Despite these limitations, glossing remains the dominant strategy for building sign language datasets, and it is the representation that LSU-446 inherits from its source corpus.

ISLR paradigms. Early systems operated on raw pixels with 3D convolutional networks such as C3D (Tran et al., 2015) and I3D (Carreira and Zisserman, 2017), which are data-hungry and computationally heavy. An alternative represents the signer as body, hand and face *keypoints* extracted by a pretrained pose estimator such as OpenPose (Cao et al., 2021) or MediaPipe (Lugaresi et al., 2019), discarding appearance and drastically reducing dimensionality. This is the natural choice in a low-resource setting, as it delegates feature extraction to a well-trained external model. Over skeletal input, graph convolutional networks modeling joints as nodes and bones as edges (ST-GCN, Yan et al., 2018; MSG3D, Liu et al., 2020) have proven effective, as have Transformers: SPOTER (Boháček and Hrúz, 2022) adapts the Transformer (Vaswani et al., 2017) to pose sequences and is competitive on WLASL (Li et al., 2020a) and LSA64 (Ronchetti et al., 2023) at low cost. We adopt SPOTER and MSG3D as baselines, contrasting attention-based and graph-based paradigms.

Temporal segmentation and sign spotting. Sign spotting addresses the problem of localizing instances of a given sign within continuous video, in contrast to continuous sign language recognition, whose objective is to transcribe complete utterances regardless of their specific timestamps (Koller et al., 2016; Camgöz et al., 2020). Sign spotting systems are commonly constructed by combining ISLR models with temporal segmentation mechanisms (Li et al., 2020b; Albanie et al., 2020), although the strategies adopted to couple these two components differ across the literature.

One family treats temporal segmentation as a task itself, labeling each frame as sign or transition and leaving classification to a downstream model. Adapting action-segmentation architectures such as MS-TCN (Farha and Gall, 2019), Renz et al. (2021a) segment signing with temporal convolutions, subsequently refined through changepoint-modulated pseudo-labeling (Renz et al., 2021b), while Moryossef et al. (2023) recast the problem as frame-level BIO tagging over pose features and report generalization across signers and domains. A second family has the classifier scan the stream directly via a sliding-window and require no additional model, at the cost of an explicit background class and of post-processing to suppress duplicate detections (Zuo et al., 2024).

Low-resource sign languages and transfer. Most benchmarks, for example, WLASL (Li et al., 2020a) or LSA64 (Ronchetti et al., 2023), consist of citation-form signs recorded under controlled conditions; corpora of spontaneous discourse are scarce. The closest work to ours is Garmendia et al. (2025), who apply MSG3D to isolated recognition of Catalan Sign Language (LSC) from spontaneous dialogue and monologue — like ours, continuous interpreter video rather than isolated signs. Over 105 curated classes, they report 49.24% accuracy.

3 The LSU-446 Dataset

3.1 Source Corpus

Our starting point is the gloss-annotated LSU corpus by Fojo et al. (2024): 27 videos of professional interpreters, each paired with an ELAN annotation file (Wittenburg et al., 2006), assigning to every sign a gloss and explicit start and end times. The material covers television broadcasts and narrated classic stories, uniformly recorded at 25 fps. Framing is heterogeneous — the interpreter may be centered in the foreground, displaced to one side, or inset in a small corner box — and spatial resolution varies widely. Since the recordings capture natural interpreting, signs are coarticulated, and the annotation contains periods of several minutes in which no signing occurs. Figure 1 illustrates the resulting continuity: consecutive glosses occupy adjacent intervals with no pause between them, so the temporal boundary between two signs is a matter of annotator judgement rather than a visible event.

3.2 Preprocessing Pipeline

Each video is cropped spatially to the region containing the interpreter and temporally to remove uninformative material such as commercial breaks, creating 34 segments from the original 27 videos, after discarding two videos whose resolution was too coarse. Frames are then extracted at 25 fps and body, hand and face keypoints are obtained with MediaPipe Holistic (Lugaresi et al., 2019). Missing keypoints were zero-filled. In parallel, `(gloss, start, end)` triplets are extracted from the ELAN files.

3.3 Gloss Normalization

The available annotations from the corpus are preliminary and not yet a standardized version. The raw label set contains 4,612 distinct strings over 28,093 instances, but many of them are notational variants of the signs. We apply four normalization rules, each motivated by a pattern found in the data.

Gender. Many LSU signs carry no grammatical gender, but in this preliminary annotation this had not been standardized yet, annotators assigned either a Spanish male or female word form, or sometimes a gender-neutral form, so `{MAESTRA, MAESTRO, MAESTRX, MAESTR@}` all denote one sign. We map all the gendered variants onto the neutral form, keeping the `@` as canonical.

Spacing and case. LSU function words and markers (e.g. `PT:DET`, the default determiner) are often written with different punctuation symbols or spacing, making `{PT:DET, PT :DET, PT: DET}` three different labels for the same sign. We remove the spurious spaces and uppercase all glosses.

Uncertainty markers. Annotators signalled uncertainty with ¿ and ?, either attached to a tentative gloss or standing alone where a sign could not be identified. We removed all question marks, and instances with no tentative gloss were discarded.

Fingerspelling. Sequences marked `FS:` are represented as a single `<FINGERSPELL>` token. Since fingerspelling is highly variable, it is left out of scope here, using a dedicated token to keep its occurrences detectable for future work.

	Raw	LSU-446
Unique signers	10	10
Sign instances	28,093	17,793
Unique glosses	4,612	446
Mean instances per gloss	6.1	39.9
Std. instances per gloss	21.4	57.3
Glosses with $\geq$10 inst.	505	446
% glosses with $\geq$10 inst.	10.9	100

Table 1: Corpus statistics before and after the preprocessing pipeline.

3.4 Resulting Dataset

After preprocessing, we retain only glosses with at least 10 instances in the corpus. Table 1 summarizes the resulting dataset, which we call **LSU-446**[*] after its class count. It retains 63.3% of the sign instances (17,793 of 28,093) concentrated in 9.7% of the glosses (446 of 4,612), raising the mean instances per gloss from 6.1 to 39.9. The raw corpus already had 505 glosses with at least 10 occurrences; the final count is lower because normalization merges variants previously counted separately.

Two properties distinguish LSU-446 from standard benchmarks, as Figure 2 shows. First, the distribution remains strongly long-tailed after filtering and the most frequent glosses are `PT:DET` (781 instances), `TAMBIÉN` (491) and `NO` (417) against a mean of 39.9, and the standard deviation *increases* from 21.4 to 57.3, since filtering removes the mass of near-empty classes and leaves a more dispersed distribution over the survivors. We therefore report both global and macro-averaged accuracies in §4. Second, signs are short: 403 ms on aver-

[*]Available at https://github.com/pln-fing-udelar/lsu-446, after accepting license of use. LSU-446 includes keypoint sequences, normalized glosses, and metadata (signer_id, split, video_fps).

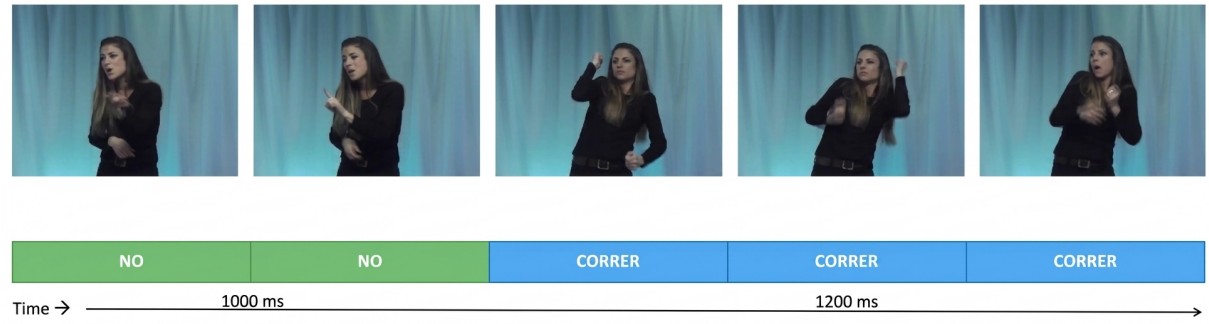


Figure 1: A sequence from LSU-446. Above, representative frames of a video; below, the temporal alignment of the corresponding glosses, at the transition from NO (not) to CORRER (to run). Consecutive signs abut directly, without pauses or a return to a neutral rest pose.

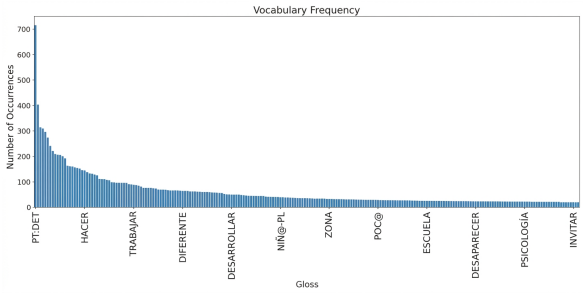


Figure 2: Instances per gloss in LSU-446, sorted by frequency (one gloss labeled every 20 on the x axis). A few glosses dominate while most lie close to the ten-instance threshold, which is why we report balanced alongside global accuracy.

age, around ten frames at 25 fps, against 2.41 s in WLASL (Li et al., 2020a). Each sign is represented by few frames, making recognition markedly more sensitive to temporal variation and to imprecise annotation boundaries.

This dataset, split as 80/10/10 at the level of individual sign instances, is used for the ISLR experiments shown in §4, but we use a different held-out video as described in §5.3 to evaluate sign spotting systems.

4 Isolated Sign Language Recognition

4.1 Models and Setup

We compare two keypoint-based architectures representing different paradigms. **SPOTER** (Boháček and Hrúz, 2022) is a Transformer over pose sequences. At 5.92M parameters it is light enough for a corpus of our size, and its preprocessing normalizes poses to universal body proportions and hands to local coordinates, mitigating differences in signer build and camera framing. **MSG3D** (Liu et al., 2020) is a graph convolutional network over the skeleton, with joints as nodes and bones as edges. Its G3D module treats space and time as a single graph instead of modelling them separately, and aggregates over several neighbourhood scales at once.

Model	Vocab.	T-1	T-5	BT-1	BT-5
SPOTER	LSU-446	24.1	42.7	20.7	37.0
SPOTER	LSU-100	32.0	53.1	30.5	50.3
MSG3D	LSU-446	**32.7**	**52.2**	**29.4**	**46.4**
MSG3D	LSU-100	**32.9**	**55.6**	**32.5**	**53.3**

Table 2: Isolated recognition accuracy (%) on the test split. T-1/T-5 are Top-1/Top-5, and BT-1/BT-5 their balanced (macro-averaged) counterparts. Bold marks the best configuration per vocabulary size.

We report Top-1 and Top-5 accuracy together with their balanced (macro-averaged) counterparts. Balanced accuracy matters here: given the long tail of §3, a model concentrating its predictions on frequent glosses can score well globally while failing on most classes. Besides the full vocabulary, we train on the 100 most frequent glosses — LSU-100 (≈10,000 instances) — for comparability with the WLASL-100 results obtained in SPOTER (Boháček and Hrúz, 2022), which reported Top-1 accuracy of 63.18% but with considerably longer signs recorded under controlled conditions.

4.2 Main Results

Table 2 reports both architectures across vocabulary sizes. On the full vocabulary MSG3D clearly outperforms SPOTER, reaching 32.7% Top-1 and 52.2% Top-5 against 24.1% and 42.7%, and it leads on balanced metrics as well, indicating that its advantage is not merely a matter of exploiting the frequent classes.

ID	min	σ	max	T-1	T-5	BT-1	BT-5
E1	—	—	—	24.1	42.7	20.7	37.0
E2	40	0.001	300	**24.3**	**43.0**	22.5	37.0
E3	40	0	300	23.2	41.8	**23.8**	**39.7**
E4	100	0.001	100	21.8	37.4	23.1	35.4

Table 3: SPOTER with augmentation and pruning on LSU-446 (%). **min/max** are the target bounds on samples per class, σ the noise added to augmented keypoints. E1 is the unaugmented baseline.

Vocabulary scaling. The most informative contrast is what happens as the label space grows. SPOTER degrades sharply from LSU-100 to LSU-446, losing almost eight Top-1 accuracy points (32.0 to 24.1), whereas MSG3D is essentially flat (32.9 to 32.7) despite the vocabulary more than quadrupling, with the expanded dataset including the least 346 frequent classes of the corpus. A possible explanation is MSG3D's explicit geometric prior: representing the skeleton as a graph with multi-scale spatio-temporal neighbourhoods may constrain the hypothesis space more than a generic attention model over coordinate vectors does, which could be an advantage when most classes have few examples.

Augmentation and pruning. Because SPOTER concentrated its predictions on frequent glosses, we evaluated seven augmentation and pruning regimes on LSU-446, varying the minimum and maximum number of samples per class and the level of Gaussian noise added to augmented keypoints. Table 3 reports a representative subset. The gains are marginal and inconsistent: mild augmentation (E2) improves Top-1 by 0.2 points, the variant without noise (E3) yields the best balanced metrics but the worst global accuracy of the three, and aggressive rebalancing (E4) degrades everything except balanced Top-1.

Cross-Lingual Transfer. Garmendia et al. (2025) report that initializing MSG3D from a model pretrained on the SWL-LSE Spanish Sign Language dataset improves Catalan Sign Language recognition by 6.61 Top-1 points. Since their setting closely matches ours — MSG3D over MediaPipe keypoints derived from spontaneous, unsegmented interpreter video — we replicated the procedure, fine-tuning the same SWL-LSE checkpoint on LSU with no layers frozen.

Surprisingly, the transfer yields no benefit. On LSU-446 the pretrained model reaches 32.3% Top-1 against 32.7% trained from scratch, a slight *decrease*; balanced Top-1 likewise falls from 29.4% to 28.0%. On LSU-100 it gains 0.6 points (33.5% versus 32.9%), well inside the range we observe between runs and far from the 6.61-point improvement reported for LSC.

We see two plausible explanations, and further experiments would be needed to confirm whether either holds. First, linguistic distance: LSE and LSC coexist in the same country and share substantial lexical material, whereas LSU developed independently, so the source lexicon may simply transfer less. Second, capture conditions: SWL-LSE consists of isolated signs recorded through a web application, while our data is broadcast footage with heterogeneous framing and resolution, a domain gap that the LSC corpus does not face to the same degree.

5 Sign Spotting

We now move from pre-segmented signs to continuous video, where the system must decide both *where* signs occur and *which* they are. We compare two ways of coupling an isolated classifier to that problem: an explicit temporal segmenter followed by classification, and a sliding window in which the classifier performs both roles.

5.1 The NO-SIGN Class

In order to use our ISLR classifiers in the sliding window approach, we needed to extend them to predict an extra class that could capture frames where no signing activity occurs; we call it the NO-SIGN class. We introduce it because the sliding-window pipeline below receives no boundary information and evaluates every window in the video, so it must reject non-signing periods on its own rather than relying on a segmenter to skip them. We built it by sampling subsequences from intervals between consecutive annotated glosses in the corpus, using the same duration distribution and proportion per video of real gaps, instead of being drawn uniformly at random. These new instances of the NO-SIGN class are added to the train, validation and test splits of §4, and we train new versions of both SPOTER and MSG3D models that consider this one additional class.

Adding NO-SIGN barely changes overall Top-1 accuracy — 23.8% versus 24.1% for SPOTER on LSU-446, 29.2% versus 32.7% for MSG3D — and on LSU-100 it even helps MSG3D, which reaches

its best overall configuration at 34.0% Top-1. The per-class metrics, however, are poor: SPOTER attains 7.7% precision and 3.4% recall on NO-SIGN ($F_1 = 4.7$), and MSG3D, in its best configuration, 20.0% precision and 23.7% recall ($F_1 = 21.7$). Both models therefore confuse intervals without signing for lexical glosses, which is unsurprising given that such intervals have no stable gestural structure; as we show below, this weak background rejection becomes the binding constraint on the sliding-window pipeline.

5.2 Evaluation Protocol

Evaluating a sign spotting system requires separating segmentation errors from lexical classification errors, since a system may, for example, locate a sign perfectly and label it wrongly. We therefore define a three-level protocol in which each level adds a stricter condition to the previous one.

Let S_p be a predicted segment, S_g a ground-truth segment with gloss G_g, and $f(\cdot)$ the classifier. All levels share a permissive temporal criterion: S_p is a true positive only if $|S_p \cap S_g| > 0$, that is, a single overlapping frame suffices. The proposed levels differ in the lexical requirement: **Level 1** (unlabeled temporal detection) is lexically independent; **Level 2** additionally requires $G_g \in \text{Top-5}(f(S_p))$, labeled but relaxed; and **Level 3** requires $G_g = \text{Top-1}(f(S_p))$, labeled and strict. At each level we report F_1, which measures whether the system found the sign: unlabeled at Level 1, labeled Top-5 at Level 2, and strictly labeled Top-1 at Level 3.

As we use a very permissive intersection criterion, F_1 alone says nothing about boundary quality: a system can detect every sign while cutting it badly. We therefore also report the temporal Intersection over Union, $\text{IoU}(S_p, S_g) = |S_p \cap S_g|/|S_p \cup S_g|$. Aggregating it admits two directions, one centered on the reference and one on the predictions, each associated to a different failure mode. We report their mean, the *General IoU*:

$$\text{IoU}_g = \frac{1}{2}\left(\frac{1}{|G|}\sum_{S_g \in G}\max_{S_p}\text{IoU} + \frac{1}{|P|}\sum_{S_p \in P}\max_{S_g}\text{IoU}\right) \quad (1)$$

where G and P are the sets of ground-truth and predicted segments. This bidirectional form penalizes both undetected signs, which lower the reference-centered term, and spurious predictions, which lower the prediction-centered term. At Levels 2 and 3 we assign $\text{IoU} = 0$ to segments whose class is not in $\text{Top-5}(f(S_p))$ or $\text{Top-1}(f(S_p))$ respectively.

5.3 The need for a new test set

The train, validation and test splits of §4 are drawn at the level of individual signs rather than whole videos, so every video in LSU-446 contributes different instances to all three splits. This means we cannot use any of those videos for evaluating sign spotting, which operates over continuous video, as there would be a leak of training signs into the test signal. Because of this, we compute all sign spotting results on a single video absent from LSU-446 altogether: a session of the Uruguayan Chamber of Representatives, 10 minutes and 31 seconds long, containing 730 annotated signs. It is deliberately adversarial: the interpreter appears in no other video in the corpus, so the signer is unseen; the domain is formal parliamentary discourse rather than television or narrated stories; and 33.8% of its signs are out-of-vocabulary (OOV).

We purposefully retain OOV signs in the ground truth. For temporal detection they are valid instances of linguistic activity, and excluding them would inflate Level 1. For lexical recognition they are unavoidable failures, since no architecture can predict a label outside its vocabulary. Keeping them makes the reported figures a faithful estimate of what these systems would do on genuinely unseen material, and the penalty applies similarly to every method compared.

5.4 Segmenter-Based Pipeline

The first approach uses an existing segmentation model before classifying. We use E1s of Moryossef et al. (2023), the reported best performing of their variants: a multi-stage temporal convolutional network over MediaPipe Holistic pose features that assigns each frame a BIO tag, from which segment boundaries are derived. We take the authors' pretrained weights; we also experimented with fine-tuning E1s on our own dataset but it slightly degraded the performance (88.2% versus 88.9% F_1, 31.7% versus 32.5% mean IoU on a held-out video), so we retain the original weights. Each predicted segment is then classified by the LSU-446 models of §4 trained without the NO-SIGN class, since segment boundaries are already supplied.

Evaluated alone on our data, the segmenter shows an expected asymmetry: an F_1 of 88.9% but a mean IoU of only 32.5%. It finds nearly all signing regions while systematically mis-cutting

Level	Model	Segmenter		Sliding win.	
		$\mathbf{F_1}$	$\mathbf{IoU}_g$	$\mathbf{F_1}$	$\mathbf{IoU}_g$
1	MSG3D	95.9	**43.4**	**99.8**	10.1
	SPOTER	95.9	**43.4**	99.7	8.7
2	MSG3D	**13.5**	**6.9**	6.4	2.3
	SPOTER	**15.6**	**7.4**	12.1	2.2
3	MSG3D	**7.1**	**3.9**	3.1	1.4
	SPOTER	**6.8**	**3.2**	3.1	0.3

Table 4: Sign spotting on the held-out video (%), over all 730 ground-truth signs including the 33.8% OOV. Level 1, 2, and 3 are unlabeled temporal detection, and labeled Top-5 and Top-1 recognition, respectively.

their extent. This asymmetry propagates through the whole pipeline.

5.5 Sliding Window

The second approach removes the segmenter. A window of W frames advances over the video with stride S; each position is classified, and every frame accumulates a Top-1 vote and five Top-5 votes from all windows covering it. The winning class per frame is taken as its label, subsequent streaks of frames sharing a label become segments, and streaks labeled NO-SIGN become the gaps between them. A median filter over blocks of three strides overwrites isolated disagreements, repairing spurious interruptions inside otherwise contiguous segments.

For a first method implementation, we set $W = 12$, $S = 4$, chosen against the mean sign duration in the dataset ($\approx$10 frames). The constraint $W = 3S$ follows from the median filter's design. This yields 3,943 windows over the 15,782 frames of the test video. Here the classifiers must reject non-signing frames themselves, so we use the NO-SIGN model variants from §5.

5.6 Results and Discussion

Table 4 reports both approaches at all three levels. Level 1 is identical for the two classifiers in the segmenter pipeline, as expected: segmentation is entirely determined by E1s (Moryossef et al., 2023), and the video is segmented once and reused.

A high F_1 can be an artifact. The sliding window attains an almost perfect Level 1 F_1 of 99.8%, against 95.9% for the segmenter — yet its General IoU is four times *worse* (10.1% versus 43.4%). The explanation is structural: the sliding window emits only 38 segments (31 for SPOTER) to cover 730 real signs, so each predicted segment spans many consecutive signs at once. Under a criterion that requires a single overlapping frame, such enormous segments detect nearly everything by construction, which is why F_1 approaches 100% while boundary agreement collapses. This is precisely the failure that the General IoU is designed to expose, and this is why Level 1 F_1 should never be read in isolation.

The root cause is the NO-SIGN weakness measured in §4. Since NO-SIGN is what delimits consecutive signs, a model that rarely predicts it cannot close a segment, and adjacent signs merge into long runs. The bottleneck of the sliding-window implementation in this work is thus not lexical discrimination but a suitable background rejection.

Explicit segmentation wins. The segmenter pipeline dominates at all recognition levels, comfortably winning in both F_1 and IoU at Levels 2 and 3 (see Table 4). Absolute values remain low — 15.6% F_1 at Level 2 and 7.1% at Level 3 — which is consistent with classifiers achieving 52.2% Top-5 in isolation now receiving imprecisely cut segments, compounded by 33.8% OOV that cannot be recovered by any model.

The ranking inverts under continuous input. In isolation MSG3D beats SPOTER on every metric (§4); in continuous video SPOTER leads at Level 2 in both pipelines (15.6 versus 13.5 with the segmenter, 12.1 versus 6.4 with the sliding window), while MSG3D remains marginally ahead at Level 3. That this inversion appears under both coupling strategies suggests it is a property of the models rather than of either pipeline: SPOTER's ranking of candidates seems to degrade more gracefully when boundaries are imprecise, whereas MSG3D's advantage depends on receiving well-segmented input.

The segmenter, not the classifier, is the ceiling. Restricting the IoU to correctly classified segments raises MSG3D's Level 3 figure to 54.2%. When the model does identify a sign, the segment it was given was well cut; the failures concentrate on segments that were poorly delimited in the first place. Combined with the segmenter's standalone profile — 88.9% F_1 against 32.5% IoU — this indicates that effort is better spent on temporal boundary precision than on the classifier, and it explains why our system detects signing reliably while recognising little of it.

6 Conclusion

We have presented the first computational study of Uruguayan Sign Language recognition at scale. We release LSU-446, a curated dataset of 17,793 sign instances over 446 glosses obtained through a reproducible normalization and keypoint-extraction pipeline, and establish (possibly signer-dependent) isolated recognition baselines in which MSG3D reaches 32.7% Top-1 over the full vocabulary and, unlike SPOTER, is barely affected by vocabulary growth. We then define a three-level evaluation protocol that separates temporal detection from lexical recognition, and use it to compare a segmenter-based pipeline against a sliding window on a fully unseen video. Explicit segmentation wins at every level involving recognition, while the sliding window's near-perfect detection score proves to be an artifact of emitting very few, very long segments.

Limitations and Future Work

Annotation. The gloss annotation we build on is a preliminary version whose notation was not yet standardized, which is why the normalization of §3 is necessary in the first place. Our rules unify what we could identify systematically, but residual inconsistency certainly remains, and some of it is indistinguishable from genuine lexical variation without expert review. A definitive, standardized glossing of the corpus would remove this uncertainty at its source.

Data splits. Our main results use an instance-level 80/10/10 split, so signs from the same video, signer and recording session can appear in both training and test. This inflates our isolated recognition figures relative to a signer-independent protocol, and the numbers in Table 2 should be read as an upper bound rather than as an estimate of generalization to new signers. We adopted it because the stricter video-level split left many classes absent from entire partitions and degraded results substantially. The sign spotting evaluation of §5 does not share this weakness, as its test video is disjoint from training in signer and domain.

Scale of the continuous evaluation. All spotting results come from a single 10-minute video with one interpreter in one domain. The comparisons in Table 4 are therefore point estimates without confidence intervals, and we run no significance testing; differences of one or two points between models should not be considered meaningful. The consistency of the ranking inversion across two independent pipelines is our only evidence that it could not be noise, and it is not conclusive.

No hyperparameter optimization. Both models retain the learning rate of their original publications, and we search neither it nor the window parameters of the sliding-window system, nor any decision threshold for NO-SIGN. Our numbers are consequently lower bounds on what these architectures could achieve on LSU, and the comparison between them assumes that inherited settings are equally favourable to both, which we cannot verify. A systematic search over these — particularly the sliding window's (W, S) and a decision threshold for NO-SIGN — is a direct and inexpensive next step.

Scope. Fingerspelling segments are collapsed into a single token and never recognized as another class, glosses occurring fewer than ten times are excluded, and 33.8% of the test video used for sign spotting systems is out of vocabulary. The systems we describe do not perform continuous sign language recognition: they locate and classify individual signs, and do not consider neighbour signs nor the grammatical structure of the processed utterances. These are all elements to be explored and expanded on subsequent research.

The sliding window deserves a better segmentation rule. The sliding window trails the segmenter pipeline, but much of the gap traces to a single design choice rather than to the approach itself. Boundaries arise only where NO-SIGN wins the per-frame vote, so the temporal structure of the output rests on a class both models predict with low recall. Several remedies are inexpensive and need no retraining: a threshold that favours NO-SIGN in ambiguous frames rather than requiring it to win outright; the vote histogram, discarded once the winning class is taken, in which a drop in agreement among overlapping windows is itself a boundary cue; and a duration prior, since with signs averaging some ten frames a run of hundreds under one label is unlikely.

Ethics Statement

The corpus consists of professional interpretation broadcast publicly on television and of recorded narrations, obtained from an existing annotated corpus (Fojo et al., 2024).

No deaf signers took part in designing or evaluating the systems reported here. Our evaluation is entirely automatic and measured against glosses produced by an annotation team, so it says nothing about whether the output would be useful or acceptable to the community the technology is meant to serve. The gloss-based framing itself embeds decisions about how LSU is segmented and labeled that were not made by the users of the language, and we note that participatory evaluation should precede any application built on this work.

Finally, we want to be explicit about the risk of overstating what these results support. A system recognizing roughly 7% of signs in continuous discourse is not an accessibility tool, and deploying one as if it were — in a medical, legal or educational setting — would be actively harmful, since errors would fall on the person with least power to contest them. Automatic recognition of LSU should be understood as an aid to annotation and research, and as a complement to professional human interpreters rather than a substitute for them.

References

Samuel Albanie, Gül Varol, Liliane Momeni, Triantafyllos Afouras, Joon Son Chung, Neil Fox, and Andrew Zisserman. 2020. BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Matyáš Boháček and Marek Hrúz. 2022. Sign pose-based transformer for word-level sign language recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, pages 182–191.

Necati Cihan Camgöz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2021. OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1):172–186.

João Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? A new model and the kinetics dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Luis Chiruzzo, Bruno Pérez Rodríguez, Juliana Bianchi-Madera, Martín Dutra, and Roberto Aguirre. 2026. Machine Translation between Uruguayan Sign Language (LSU) Glosses and Spanish. *Procesamiento del Lenguaje Natural*, 77.

Yazan Abu Farha and Juergen Gall. 2019. MS-TCN: Multi-stage temporal convolutional network for action segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Alejandro Fojo, Marcela Tancredi, Fabricio Etcheverry, Allison Pintos, Bruno Pérez, Juliana Bianchi, Roberto Aguirre, Mauricio Castillo, María Noel Macedo, and Luis Chiruzzo. 2024. Corpus de lengua de señas uruguaya y desarrollo de metadatos. In *Congresso Internacional de Pesquisas em Linguística de Línguas de Sinais*.

Naiara Garmendia, Horacio Saggion, and Euan McGill. 2025. Spontaneous Catalan Sign Language recognition: Data acquisition and classification. In *Proceedings of the Third International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL)*, pages 7–15, Geneva, Switzerland. European Association for Machine Translation.

Oscar Koller, Sepehr Zargaran, Hermann Ney, and Richard Bowden. 2016. Deep sign: Hybrid CNN-HMM for continuous sign language recognition. In *Proceedings of the British Machine Vision Conference (BMVC)*.

Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. 2020a. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 1459–1469.

Dongxu Li, Xin Yu, Chenchen Xu, Lars Petersson, and Hongdong Li. 2020b. Transferring cross-domain knowledge for video sign language recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6204–6213.

Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. 2020. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. MediaPipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*.

Ministerio de Salud Pública de Uruguay. 2023. Personas sordas e hipoacusia por primera vez en el Ministerio de Salud Pública.

Amit Moryossef, Zifan Jiang, Mathias Müller, Sarah Ebling, and Yoav Goldberg. 2023. Linguistically

motivated sign language segmentation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12703–12724, Singapore. Association for Computational Linguistics.

Amit Moryossef, Kayo Yin, Graham Neubig, and Yoav Goldberg. 2021. Data augmentation for sign language gloss translation. In *Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL)*, pages 1–11, Virtual. Association for Machine Translation in the Americas.

Katrin Renz, Nicolaj C. Stache, Samuel Albanie, and Gül Varol. 2021a. Sign language segmentation with temporal convolutional networks. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Katrin Renz, Nicolaj C. Stache, Neil Fox, Gül Varol, and Samuel Albanie. 2021b. Sign segmentation with changepoint-modulated pseudo-labelling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Franco Ronchetti, Facundo Manuel Quiroga, César Estrebou, Laura Lanzarini, and Alejandro Rosete. 2023. LSA64: An argentinian sign language dataset. *Preprint*, arXiv:2310.17429.

Ariel E. Stassi, Marcela Tancredi, Roberto Aguirre, Alvaro Gómez, Bruno Carballido, Andrés Méndez, Sergio Beheregaray, Alejandro Fojo, Víctor Koleszar, and Gregory Randall. 2022. LSU-DS: An uruguayan sign language public dataset for automatic recognition. In *Proceedings of the 11th International Conference on Pattern Recognition Applications and Methods (ICPRAM)*, pages 697–705. SciTePress.

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3D convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Peter Wittenburg, Hennie Brugman, Albert Russel, Alex Klassmann, and Han Sloetjes. 2006. ELAN: A professional framework for multimodality research. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*, pages 1556–1559.

Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Xuan Zhang and Kevin Duh. 2021. Approaching sign language gloss translation as a low-resource machine translation task. In *Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL)*, pages 60–70.

Ronglai Zuo, Fangyun Wei, and Brian Mak. 2024. Towards online continuous sign language recognition and translation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11050–11067, Miami, Florida, USA. Association for Computational Linguistics.

Coverage Is Not Enough: A Controlled Study of Annotation Selection for Low-Resource Sign Language

Gyanendra Shrestha
Florida State University
gshrestha@fsu.edu

Liting Zhang
University of West Florida
lzhang@uwf.edu

Abstract

Expert annotation is a central bottleneck in sign-language corpus construction. A natural assumption is that annotation budgets should prioritize diverse or underrepresented signing patterns. We test this in a controlled retrospective study on PHOENIX14T (German Sign Language) and Isharah (Saudi Sign Language), across gloss recognition, translation, multiple budgets, and three seeds, comparing random sampling against geometric, active-learning, pose-derived articulatory, and privileged selectors. Articulatory coverage is generally weak across all four corpus–task settings and rarely outperforms random sampling. A density analysis shows it selects locally atypical clips even after controlling for duration; removing the sparsest 10–20% helps at larger budgets but never enough to beat random. A privileged gloss-diversity oracle also performs poorly despite selecting comparatively typical clips, showing that atypicality is not the whole explanation. Facility-location representativeness is the strongest feasible selector in three of four largest-budget settings, with lower WER than random in 16 of 18 seed-matched recognition comparisons, compared with only 2 of 18 for articulatory-only selection. The results suggest pose-derived articulatory coverage and gloss-type coverage, as operationalized here, are unreliable proxies for annotation utility in low-resource sign-language corpus construction. The source code is available at https://github.com/gyan-shrestha/WSLP-2026.

1 Introduction

Continuous sign-language corpora are expensive to annotate because glossing, translation, segmentation, and temporal alignment require specialist expertise. Recorded video can therefore substantially exceed manually annotated material: Content4All released approximately 190 hours of broadcast signing, but only about 20 hours received manual alignment by Deaf experts and interpreters (Camgöz et al., 2021). This bottleneck is particularly consequential for underrepresented sign languages (De Sisto et al., 2022).

A corpus team with a fixed budget must decide which clips to annotate before their linguistic content can be observed through annotation. Active learning selects uncertain examples using an existing model, while core-set and batch methods seek geometric diversity or representativeness (Sener and Savarese, 2017; Ash et al., 2019). Sign-language studies have explored active selection for isolated-sign acquisition and fingerspelling (Bicego et al., 2023; Wang and Nalisnick, 2023), but selection from continuous unlabeled video before a reliable recognizer or gloss inventory exists remains underexplored.

Sign languages suggest a natural label-free alternative. Hand configuration, location, movement, coordination, and non manual activity can be approximated from pose without manual labels (Börstell, 2023; Kezar et al., 2023). One might therefore spend the annotation budget on clips that add underrepresented articulatory patterns. The intuition is appealing, but coverage need not translate into annotation utility, measured here as downstream task performance under a fixed budget: learning may benefit more from repeated observations of common, representative structures than from maximizing novelty.

We test this hypothesis on PHOENIX14T (Camgoz et al., 2018) and Isharah (Alyami et al., 2026), hiding labels during selection and revealing them only for purchased clips. Pose-derived articulatory coverage, as operationalized here, is generally weak across both sign languages and both tasks, rarely outperforming random sampling. We then ask why. Coverage systematically favors locally atypical clips, yet removing the sparsest candidates recovers only part of the performance gap. A privileged gloss-diversity selector provides the com-

Proceedings of the Second Workshop on Sign Language Processing, pages 121–131
October 29, 2026 ©2026 Association for Computational Linguistics

plementary case: it performs poorly even while selecting comparatively typical clips. The evidence therefore points to a broader mismatch between the coverage objectives evaluated here and downstream annotation utility, rather than a failure of pose representation alone.

2 Controlled Study

2.1 Retrospective annotation protocol

Let the unlabeled pool be $\mathcal{U} = \{u_1, \ldots, u_N\}$, with estimated annotation cost c_i for clip u_i. Each selector chooses a subset whose total cost does not exceed budget B. Labels remain inaccessible until selection is complete. PHOENIX14T contains 7,096 training clips from studio weather broadcasts; Isharah contributes a 10,000-clip signer-independent pool recorded with smartphones in varied settings. All budgets use a common 6× duration-based annotation-time multiplier, which holds the cost convention fixed across tasks (Appendix G gives the exact hour-to-percent mapping). Recognition budgets span 5 to 20 percent of pool cost: 2.8/5.5/11.0 annotation hours for PHOENIX14T and 7.1/14.2/28.4 for Isharah. Translation follows the same schedule for Isharah and a finer 0.5/1/2/5-hour sweep for PHOENIX14T. Clip cost is proportional to duration, and each budget is defined as a fixed fraction of the total pool cost. Changing the recognition rate from 6× to 40× therefore rescales the reported annotation hours but does not change which clips are affordable, their greedy gain-per-hour ordering, or the subsets ultimately selected. The experimental results are therefore unchanged under this alternative rate (Appendix G). Because cost is duration-based, selectors that prefer shorter clips purchase more clips at the same budget (Table 10). These budgets target severe low-resource conditions in which selection strategy can substantially affect the composition of the labeled subset.

2.2 Selection strategies

Baselines and geometric selection. Random sampling selects clips until the budget is exhausted. Source-stratified random allocates budget across recording sources first, then samples within each; it is a source-imbalance control rather than a primary strategy, so it appears in Appendix A rather than Table 1. For geometric selection, each normalized pose sequence is projected to a generic embedding z_i: greedy k-center spans the embedding space (Sener and Savarese, 2017), while facility location favors clips similar to many others, emphasizing dense, representative regions.

Articulatory coverage. Our selection objective favors clips that add signing patterns not yet well represented in the labeled set. Because labels are unavailable during selection, we approximate six articulatory channels from pose: hand configuration, orientation, location, movement, two-hand coordination, and non-manual activity. Each channel is clustered into a separate discrete codebook, so each clip can be represented by the codewords assigned to its frames. Selection can then favor clips that introduce articulatory patterns that are missing or underrepresented in the current labeled set.

The objective incorporates three considerations. First, repeated observations of the same codeword should have diminishing value, so additional coverage contributes progressively less. Second, rare codewords should receive greater weight because common patterns are easier to acquire. Third, clips with poor pose tracking should contribute less because their assigned codewords are more likely to reflect measurement error. We capture these effects with a logarithmic coverage term, a codeword rarity weight r_{gk}, and a clip-level pose-quality score q_i:

$$F_{\text{art}}(\mathcal{S}) = \sum_g \frac{\omega_g}{K_g} \sum_{k=1}^{K_g} r_{gk}\, \ell_{gk}(\mathcal{S}),$$

$$\ell_{gk}(\mathcal{S}) = \log\left(1 + \sum_{u_i \in \mathcal{S}} q_i n_{igk}\right),$$

where K_g is the number of codewords in channel g, ω_g is its channel weight, and n_{igk} is the capped number of frames in clip i assigned to codeword k. The logarithm gives diminishing returns and makes F_{art} submodular, which supports greedy optimization. The rarity weight r_{gk} increases as a codeword becomes less frequent in the pool, while q_i is floored at 0.1 so that poorly tracked clips are down-weighted rather than excluded. Closed-form definitions of both weights are provided in Appendix G.

We evaluate three nested selectors based on this objective: *articulatory only*, which uses F_{art} alone; *articulatory + repr.*, which adds the facility-location term F_{rep}; and the *full objective*, $0.60F_{\text{art}} + 0.30F_{\text{rep}} + 0.10F_{\text{src}}$, which also rewards coverage across recording sources. The weights are fixed a priori and are not tuned on any evaluation set. All three selectors use greedy selection by marginal gain per purchased hour.

Model-dependent and privileged comparisons. To separate a weakness of the coverage objective from one of the pose representation, we also compare against selectors with stronger signals. Entropy sampling and BADGE (Ash et al., 2019) score the pool with a recognizer trained on a random seed set costing 2% of the target annotation budget, making them feasible but not label-free. Privileged diagnostics read labels unavailable at selection time: text facility location operates on hidden translations, and the gloss oracle favors gloss types not yet covered, testing whether coverage helps when defined over linguistic content rather than pose-derived codewords.

Density diagnostic. We measure each clip's mean cosine distance to its 20 nearest neighbors in the generic pose embedding; larger values indicate sparser local regions. To control for duration, we compare selector density within duration quartiles. We then rerun articulatory coverage after excluding the sparsest 10% or 20% of candidates, leaving the objective, greedy procedure, and budget unchanged.

2.3 Tasks and evaluation

Gloss recognition uses a lightweight pose model with a 256-dimensional projection, temporal convolution, Transformer encoding, and CTC output. Translation uses a matched pose encoder with an autoregressive text decoder. Within each setting, all selectors share the same architecture, optimization, and decoding settings; Appendix G gives full implementation details. The recognition vocabulary contains only glosses present in the purchased subset: the CTC output layer can only emit trained-on glosses, so a reference gloss absent from the purchased subset is scored as an error at evaluation time, a deliberate measure of one consequence of the annotation-selection decision. We report word error rate (WER; lower is better) for recognition and BLEU (higher is better) for translation, with mean and standard deviation over three seeds; privileged diagnostics use one run. The study concerns selector ranking under severe label scarcity rather than state-of-the-art absolute accuracy.

3 Results

3.1 Articulatory coverage is generally weak

Table 1 summarizes the largest-budget results and Figure 1 shows every recognition budget as a difference from random; Appendix A gives the underlying absolute numbers. On PHOENIX14T recognition, WER rises from 77.46 with representativeness to 78.24 with articulatory + representativeness, 78.90 with the full objective, and 81.50 with articulatory-only selection, the same ordering as at 2.8 and 5.5 hours. PHOENIX14T translation is similarly unfavorable: at 5 hours, representativeness reaches 5.68 BLEU, compared with 4.69 for the full objective and 3.84 for articulatory-only selection. Across the six recognition corpus–budget settings (three PHOENIX14T budgets, three Isharah budgets), articulatory-only selection never outperforms random sampling (Appendix A); on translation it wins only once, at the largest Isharah budget.

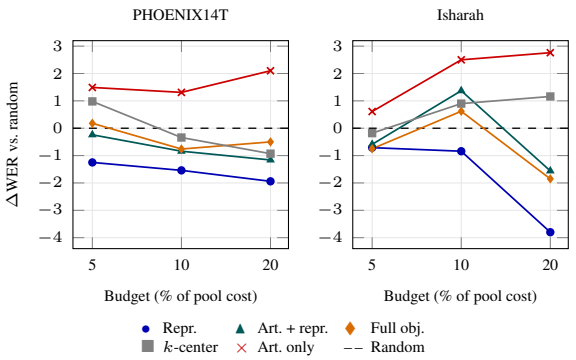


Figure 1: Recognition WER relative to random sampling at each budget, where negative values indicate lower WER than random and the dashed line denotes the random baseline. Articulatory-only selection has higher mean WER than random at every budget on both corpora, while adding representativeness lowers mean WER in each case. Representativeness alone has lower mean WER than random at every budget. Values are computed from Appendix Tables 3 and 4. The privileged gloss oracle is omitted because it falls outside the plotted range, reaching up to $+9.7$ WER relative to random on Isharah; see Table 1.

The pattern transfers to Isharah recognition. At 28.4 hours, representativeness obtains 76.24 WER, versus 78.19 for the full objective and 82.80 for articulatory only. At the smallest budget, the strongest methods are effectively tied, but the separation grows as more data are purchased. Active learning (entropy sampling, BADGE) and coreset selection (k-center) are also inconsistent across budgets, whereas facility-location representativeness is more reliable: across the 18 corpus–budget–seed comparisons on recognition, computed from individual per-seed WER values (Appendix Table 5), it has lower WER than random in 16 cases against articulatory-only's 2. Because runs are matched by seed, we compare the 18 paired WER

Strategy	PHX CSLR WER ↓ (11.0 h)	Isharah CSLR WER ↓ (28.4 h)	PHX SLT BLEU ↑ (5 h)	Isharah SLT BLEU ↑ (28.4 h)
Representativeness	**77.46 ± 0.2**	**76.24 ± 2.0**	**5.68 ± 0.2**	6.75 ± 0.2†
k-center	78.47 ± 0.8	81.20 ± 2.2	2.84 ± 0.1	**6.93 ± 0.8**
Articulatory + repr.	78.24 ± 0.8†	78.48 ± 1.2†	4.62 ± 0.3	5.66 ± 0.3
Full obj. (art+repr+src)	78.90 ± 0.6	78.19 ± 1.7†	4.69 ± 0.5	6.90 ± 0.4†
Random	79.40 ± 1.5	80.04 ± 3.0†	4.40 ± 0.5	5.40 ± 0.9†
Articulatory only	81.50 ± 0.3	82.80 ± 0.6	3.84 ± 0.6	5.88 ± 0.4†
Text facility location (priv.)	79.19	*75.13*	3.81	3.79
Gloss coverage oracle (priv.)	85.33	89.77	3.89	2.88

Table 1: Results at the largest budget in each setting. PHX denotes PHOENIX14T. The three articulatory rows are nested: articulatory coverage alone, then + representativeness, then + source coverage, corresponding to the full objective with weights $0.60/0.30/0.10$. Bold indicates the numerically best feasible selector by mean and does not imply statistical significance. † indicates a mean whose ± 1 standard-deviation interval over three seeds overlaps that of the best method; unmarked rows do not overlap the best under this criterion. For Isharah SLT, every method except articulatory + repr., including random, falls within this range of the best, so the leading methods cannot be reliably ranked. Italic indicates an oracle diagnostic that requires labels and is unavailable during unlabeled selection. Entropy sampling and BADGE are feasible baselines reported in Appendix A.

differences directly. Representativeness outperforms random under both tests, with a two-sided sign-test $p = 0.0013$ and Wilcoxon signed-rank $p < 10^{-4}$, corresponding to a mean WER reduction of 1.68 points. Articulatory-only shows the opposite pattern, with higher WER than random under both the sign test ($p = 0.0013$) and Wilcoxon signed-rank test ($p = 0.00025$), with a mean increase of 1.80 points. These 18 pairs are not fully independent because they reuse three seeds across nested budgets. We therefore use the paired tests to summarize the consistency of the observed win counts, rather than as evidence from 18 independent experimental replicates.

3.2 Coverage is not annotation utility

A weak pose representation could explain the pattern above on its own, so the next diagnostic removes pose from the selector entirely and substitutes exact lexical identity. At the 11-hour PHOENIX14T budget the privileged gloss oracle acquires 1,085 of 1,089 gloss types yet yields 85.33 WER, compared with 79.40 for random and 77.46 for representativeness. This shows that lexical type coverage, like articulatory coverage, need not correspond to annotation utility; the failure is not specific to pose descriptors. On Isharah, oracle WER is 89.77, 9.73 points worse than random. Even with exact lexical categories, maximizing type breadth can sacrifice the frequency, context, and repetition a type needs to be learned from.

The selected subsets show the same disconnect (Appendix Table 10, which also reports clip counts). At the 10% PHOENIX14T budget, articulatory coverage acquires more gloss types than representativeness but produces higher WER, while representativeness purchases more mutually similar clips. Breadth and redundancy therefore need not track annotation utility.

3.3 Representativeness is strongest in three of four settings

Facility-location representativeness has the best mean result for recognition on both corpora and for PHOENIX14T translation. Isharah translation is less conclusive: at the largest budget, k-center, the full objective, and representativeness obtain 6.93, 6.90, and 6.75 BLEU, respectively, while random obtains 5.40 ± 0.9. Under the descriptive variability criterion in Table 1, all of these methods overlap the best method's range, whereas articulatory + repr. does not. The smaller-budget BLEU values are also low relative to their run-to-run variation. We therefore do not assign a clear ranking among the leading methods for Isharah translation.

4 Why Does Coverage Fail?

Coverage favors locally atypical clips. Articulatory-only selection (Table 2) is sparser than the pool in every duration quartile of both corpora, most clearly on Isharah, while random stays near baseline: coverage systematically favors unusual pose configurations rather than inheriting a duration bias.

Filtering helps but does not rescue coverage. If atypicality were the sole reason coverage fails, removing the sparsest candidates should eliminate

	Duration-controlled Δ kNN distance		Articulatory-only WER at largest budget		
Corpus	Art. coverage	Random	Unfiltered	Best filtered	Random
PHOENIX14T	$[+0.017, +0.034]$	$[-0.009, +0.007]$	81.50	79.94	79.40
Isharah	$[+0.058, +0.092]$	$[-0.006, +0.014]$	82.80	81.99	80.04

Table 2: Density and filtering diagnostics. Δ kNN distance is given as the range over the four duration quartiles, each measured against its own quartile pool mean; positive values indicate locally sparser clips. Articulatory coverage is sparser than the pool in every quartile of both corpora, whereas random straddles zero. The WER columns report articulatory-only selection before filtering, after removing the sparsest 10–20% of candidates, and random for reference.

the deficit. Instead it produces only partial recovery: articulatory-only WER improves from 81.50 to 79.94 on PHOENIX14T and 82.80 to 81.99 on Isharah, yet the best filtered variant remains at or above random's mean WER in all six recognition settings (Appendix Table 9). Atypical-example selection therefore contributes to the weakness of coverage but does not fully explain it.

Diversity can fail even without atypicality. The privileged gloss oracle is a complementary counterexample: on PHOENIX14T it is the weakest recognition selector despite being denser than the pool in three of four quartiles (Appendix Table 8) and acquiring nearly the complete gloss inventory. Even direct access to lexical categories can sacrifice the repetitions or combinations needed for learning.

Supporting diagnostics. Two additional diagnostics support this interpretation (Appendix D): random selection saturates articulatory codebooks quickly, and coverage-based selection is more sensitive to pose corruption than representativeness is. The learned channels also show low signer leakage and strong temporal coherence (Appendix E), arguing against the explanation that the representation is dominated by signer identity or frame-level noise. Together, these diagnostics point to a mismatch between the coverage objectives evaluated here and annotation utility, rather than a single implementation failure.

5 Discussion

The results highlight a distinction between coverage and learnability. A selector can acquire many rare articulatory or lexical types without purchasing enough repeated, representative evidence for those types to be learned effectively. This helps explain why filtering atypical clips improves articulatory coverage only partially, and why even near-complete lexical coverage can perform poorly. Facility-location representativeness provides the strongest results in three of the four corpus–task settings, while Isharah translation does not yield a clear ranking among the leading methods.

These results are specific to the low-resource, pose-based setting evaluated here. Both downstream models are small (4.1M parameters), so the observed ranking reflects the value of different selection strategies under a fixed annotation budget for models of this scale. Larger models, particularly those operating on RGB input, may be able to exploit forms of diversity that these models cannot, and a coverage-selected subset could therefore rank differently in that setting. We therefore interpret our results as evidence that coverage is not a reliable default for low-resource, pose-based annotation selection, rather than as a claim that coverage can never be beneficial.

6 Conclusion

Selecting clips to maximize pose-derived coverage or lexical type coverage, as operationalized by the objectives tested here, is not reliably better than random sampling in low-resource sign-language annotation. Across two corpora and two tasks, pose-based coverage underperforms, and privileged lexical coverage does not rescue the underlying idea. This does not imply that all diversity objectives fail; our conclusion is limited to the coverage objectives evaluated here. Selection policies should therefore be justified by annotation utility, with representativeness treated as a strong empirical baseline rather than coverage as a default objective.

Limitations

The study uses sentence-segmented benchmarks rather than unsegmented community corpora, and fixed budget units rather than a prospective annotator study. Segmentation is also part of the data collection process, not just a fixed property of the benchmark. Our selectors operate on clips

with predefined boundaries, whereas a corpus built from continuous recordings must first decide where those boundaries fall. Coverage-based selection may behave differently when segmentation itself is part of the selection process. Both downstream models are small and pose-based, so conclusions concern selector ranking under severe label scarcity and may not transfer to large RGB models (see Section 5). PHOENIX14T is a narrow weather domain, Isharah provides only one additional sign language, and the gloss oracle tests type coverage rather than all diversity objectives.

The pose pipeline also exposes a measurement-quality concern: dominant-hand detection varies substantially across signers, with a 0.365 spread on Isharah, so a noise-sensitive selector may allocate annotation partly by measurement quality rather than signing content. We do not infer a causal effect on translation; the signer-level correlation with BLEU is not statistically significant ($\rho = 0.42$, $p = 0.27$, $n = 9$).

All seed-level results use three seeds. This is too few for reliable statistical testing of individual cells in Table 1, so small differences are described as within run-to-run variation rather than assigned per-cell significance tests. Section 3.1 instead uses the seed-matched design to analyze the 18 paired differences across conditions. These paired comparisons provide more information than any single three-seed cell, but they are still not fully independent because the same seeds are reused across nested budgets. We therefore interpret the resulting tests as evidence about the consistency of the paired differences, not as inference from 18 independent replicates. The $6\times$ multiplier used to price the annotation budgets (Appendix G) is an assumption rather than a measured glossing rate. Although the multiplier does not affect clip selection or any within-task comparison, it does affect how the reported hours should be interpreted across tasks. If glossing requires approximately $40\times$ real time while translation requires $6\times$, then recognition and translation experiments reported at the same nominal number of hours do not correspond to the same annotator effort. The hour budgets should therefore be interpreted within each task rather than as directly comparable annotation costs across tasks.

Acknowledgments

We thank anonymous reviewers for their valuable feedback.

References

Sarah Alyami, Hamzah Luqman, Sadam Al-Azani, Maad Alowaifeer, Yazeed Alharbi, and Yaser Alonaizan. 2026. Isharah: A large-scale multi-scene dataset for continuous sign language recognition. *IEEE Transactions on Multimedia*.

Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. 2019. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*.

Manuele Bicego, Manuel Vázquez-Enríquez, and José L Alba-Castro. 2023. Active class selection for dataset acquisition in sign language recognition. In *International Conference on Image Analysis and Processing*, pages 304–315. Springer.

Carl Börstell. 2023. Extracting sign language articulation from videos with mediapipe. In *Proceedings of the 24th nordic conference on computational linguistics (nodalida)*, pages 169–178.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7784–7793. Ieee.

Necati Cihan Camgöz, Ben Saunders, Guillaume Rochette, Marco Giovanelli, Giacomo Inches, Robin Nachtrab-Ribback, and Richard Bowden. 2021. Content4all open research sign language translation datasets. In *2021 16th IEEE international conference on automatic face and gesture recognition (FG 2021)*, pages 1–5. IEEE.

Mirella De Sisto, Vincent Vandeghinste, Santiago Egea-Gómez, Mathieu De Coster, Dimitar Shterionov, and Horacio Saggion. 2022. Challenges with sign language datasets for sign language recognition and translation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2478–2487.

Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro-i Nieto. 2021. How2sign: a large-scale multimodal dataset for continuous american sign language. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2734–2743. IEEE.

Lee Kezar, Jesse Thomason, and Zed Sehyr. 2023. Improving sign recognition with phonology. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2732–2737.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. 2019. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*.

Ozan Sener and Silvio Savarese. 2017. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*.

Shuai Wang and Eric Nalisnick. 2023. Active learning for multilingual fingerspelling corpora. *arXiv preprint arXiv:2309.12443*.

A Complete Results at All Budgets

Table 1 in the main text reports the largest budget in each setting. Tables 3 to 7 report results at every budget.

Strategy	2.8 h	5.5 h	11.0 h
Representativeness	**88.73** ±0.5	**85.05** ±0.3	**77.46** ±0.2
Articulatory + repr.	89.74 ±0.6	85.75 ±0.4	78.24 ±0.8
k-center	90.96 ±0.5	86.25 ±0.6	78.47 ±0.8
Full without quality wt.	89.68 ±1.2	85.54 ±0.1	78.42 ±0.4
Full obj. (art+repr+src)	90.16 ±0.5	85.83 ±0.2	78.90 ±0.6
Full without rare wt.	90.43 ±0.5	86.10 ±0.6	79.08 ±0.4
Random	89.98 ±1.2	86.59 ±0.8	79.40 ±1.5
BADGE	89.91 ±1.7	85.87 ±0.7	79.71 ±0.8
Source-stratified random	89.93 ±0.6	87.66 ±1.0	81.13 ±1.1
Entropy sampling	–	87.10 ±0.8	81.23 ±0.8
Articulatory only	91.47 ±0.8	87.90 ±1.1	81.50 ±0.3
Text facility loc.	*88.50*	*85.11*	*79.19*
Gloss oracle	*92.98*	*89.57*	*85.33*

Table 3: PHOENIX14T gloss recognition, WER ↓. Entropy sampling is defined at the 10 and 20 percent budgets only.

Strategy	7.1 h	14.2 h	28.4 h
Representativeness	90.78 ±0.5	**88.15** ±0.3	**76.24** ±2.0
Full obj. (art+repr+src)	**90.74** ±0.6	89.61 ±0.3	78.19 ±1.7
Articulatory + repr.	90.90 ±0.3	90.36 ±0.7	78.48 ±1.2
BADGE	91.03 ±0.4	89.84 ±1.5	79.45 ±1.1
Source-stratified random	91.61 ±1.5	89.63 ±0.3	79.70 ±4.1
Random	91.49 ±0.4	88.99 ±0.6	80.04 ±3.0
k-center	91.31 ±0.9	89.89 ±0.4	81.20 ±2.2
Entropy sampling	–	89.82 ±2.3	82.40 ±4.3
Articulatory only	92.10 ±0.9	91.49 ±1.0	82.80 ±0.6
Text facility loc.	*90.37*	*89.06*	*75.13*
Gloss oracle	*91.56*	*90.98*	*89.77*

Table 4: Isharah gloss recognition, WER ↓. At 7.1 h the strongest three strategies lie within 0.2 WER and should be read as tied.

B Quartile-Level Density by Selector

Section 4 states that the gloss oracle is denser than the duration-matched pool in three of four quartiles on PHOENIX14T. Table 8 gives the full per-quartile breakdown behind that claim and behind Table 2, for every selector discussed in the main text, not only articulatory-only and random.

Budget	Seed 0	Seed 1	Seed 2
PHOENIX14T, representativeness − random (WER)			
2.8 h	−0.29	−1.65	−1.79
5.5 h	−0.51	−2.64	−1.47
11.0 h	+0.05	−2.83	−3.04
Isharah, representativeness − random (WER)			
7.1 h	−0.81	−0.54	−0.79
14.2 h	−1.19	+0.21	−1.52
28.4 h	−2.46	−4.95	−3.97
PHOENIX14T, articulatory-only − random (WER)			
2.8 h	+2.16	+1.41	+0.91
5.5 h	+0.93	+1.52	+1.49
11.0 h	+3.58	+1.55	+1.17
Isharah, articulatory-only − random (WER)			
7.1 h	+1.82	−0.13	+0.13
14.2 h	+2.07	+4.28	+1.17
28.4 h	+5.28	−1.36	+4.36

Table 5: The 18 corpus–budget–seed comparisons behind the win counts in Section 3.1, as (selector WER − random WER); negative means the selector beats random at that seed. From the per-seed runs underlying Tables 3 and 4.

Strategy	0.5 h	1 h	2 h	5 h
Representativeness	**4.01** ±0.1	**4.08** ±0.2	**4.49** ±0.2	**5.68** ±0.2
Random	2.59 ±0.3	3.87 ±0.5	3.57 ±0.5	4.40 ±0.5
Articulatory + repr.	1.99 ±0.2	3.57 ±0.1	3.84 ±0.4	4.62 ±0.3
Full obj. (art+repr+src)	1.60 ±0.2	2.62 ±0.1	3.06 ±0.2	4.69 ±0.5
Articulatory only	0.93 ±0.1	1.12 ±0.2	1.91 ±0.2	3.84 ±0.6
k-center	0.87 ±0.2	1.38 ±0.1	1.85 ±0.2	2.84 ±0.1

Table 6: PHOENIX14T translation, BLEU ↑.

C What Each Strategy Purchases

Table 10 describes the selected subsets at the 10-percent PHOENIX14T budget rather than the models trained on them.

Articulatory coverage acquires *more* gloss types than representativeness, 497 against 451, and still produces higher error. At this 10-percent (5.5-hour) budget the gloss oracle acquires 1,024 of 1,089 types, fewer than at the 20-percent (11-hour) budget reported in Section 3.2 (1,085 of 1,089), and is the weakest method overall at both. Meanwhile the strongest selector is one of the most redundant methods, 0.527 against 0.448 for random: it deliberately purchases clips that resemble one another. Type breadth and redundancy both point the opposite way from annotation utility.

Representativeness acquires fewer distinct types than articulatory-only selection, but provides more repeated evidence for each one: 13.37 occurrences per type on average, compared with 11.33. In other words, representativeness favors depth, whereas articulatory coverage favors breadth. The gloss or-

Strategy	7.1 h	14.2 h	28.4 h
k-center	0.92 ±0.2	1.87 ±0.4	**6.93** ±0.8
Full obj. (art+repr+src)	1.19 ±0.3	**2.66** ±0.5	6.90 ±0.4
Representativeness	1.17 ±0.4	1.92 ±0.5	6.75 ±0.2
Articulatory only	**1.24** ±0.5	2.35 ±0.5	5.88 ±0.4
Articulatory + repr.	1.12 ±0.2	2.11 ±0.3	5.66 ±0.3
Random	**1.24** ±0.2	2.56 ±0.5	5.40 ±0.9

Table 7: Isharah translation, BLEU ↑. Discussed in Section 3.3; scores below 14.2 h have standard deviations too large relative to the spread to interpret the ordering.

Strategy	Q1	Q2	Q3	Q4
PHOENIX14T				
Articulatory only	+0.017	+0.034	+0.025	+0.022
Full obj. (art+repr+src)	+0.006	+0.022	-0.002	-0.007
Representativeness	+0.012	-0.003	-0.104	–
k-center	+0.012	+0.116	–	–
Gloss oracle (priv.)	-0.007	+0.003	-0.017	-0.021
Random	+0.004	-0.007	+0.007	-0.009
Isharah				
Articulatory only	+0.058	+0.058	+0.061	+0.092
Full obj. (art+repr+src)	+0.035	+0.046	+0.060	+0.041
Representativeness	+0.049	+0.025	-0.015	-0.075
k-center	+0.044	+0.167	–	–
Gloss oracle (priv.)	+0.035	+0.035	+0.042	+0.076
Random	+0.014	+0.009	-0.006	-0.002

Table 8: Mean kNN distance in the generic pose embedding, expressed as a difference from each quartile's own pool mean, for every selector's purchases at the 10-percent budget. Positive values indicate clips that are sparser than typical clips of the same duration. – – marks a quartile with fewer than 10 purchased clips.

acle shows the same trade-off at the extreme. It covers 1,024 of 1,089 types, yet provides only 6.06 tokens per type, roughly half the support of any feasible selector, and is also the weakest method for recognition despite its near-complete type coverage. The same pattern appears at the 20-percent budget, where representativeness provides 20.16 tokens per type compared with 17.38 for articulatory-only selection. These results suggest that coverage alone is insufficient: recognition also benefits from repeated evidence for the types that are selected.

D Supporting Diagnostics

We use two additional diagnostics to examine the behavior of articulatory coverage. The saturation experiment varies the hand-configuration codebook size as a diagnostic sweep. The main hand-configuration channel used everywhere else in the paper, including every row of Table 1, has 64 codewords; the 40, 150 and 400 settings below are diagnostic sweeps only and are not used by the main selector. The corruption experiment measures how stable each selected subset remains when detected

Budget	Unfiltered	Dens10	Dens20	Random
PHOENIX14T				
2.8 h	91.47	91.81	91.94	89.98
5.5 h	87.90	87.73	86.92	86.59
11.0 h	81.50	79.94	80.19	79.40
Isharah				
7.1 h	92.10	93.20	91.61	91.49
14.2 h	91.49	90.21	90.05	88.99
28.4 h	82.80	81.99	82.02	80.04

Table 9: Articulatory-only WER (mean over three seeds) before filtering, after removing the sparsest 10% and 20% of candidates, and for random selection, at all six recognition budgets. The best filtered variant remains at or above random's mean WER in every setting.

hand landmarks are randomly dropped.

E Validating the Articulatory Channels

Any codebook fitted to pose produces clusters. Whether those clusters separate articulation rather than the people producing it is a separate question, and the failure mode is concrete: clustering handshape descriptors can converge on hand size, camera distance or seating position. Such clusters would still yield a coverage curve, so the problem is invisible downstream.

We measure the excess normalized mutual information between cluster assignment and signer identity, relative to a permutation null. Raw NMI is biased upward when clusters are many and signers few, so the null distribution is the only meaningful reference. We also test whether cluster identity persists across adjacent frames, as articulatory state should, and whether the handshape and location channels are redundant.

The thresholds are calibrated on synthetic pose where the answer is known by construction: one set in which handshape is drawn independently of signer, and one in which handshape is a deterministic function of signer. Without this calibration a passing score cannot be interpreted. The 0.05 signer-leakage threshold sits roughly an order of magnitude above the independent-control value (0.007) and well below the signer-determined one (0.698), so both corpora's observed values (0.018, 0.037) are unambiguously on the independent side.

F Pose Reliability Across Signers

The measurement-disparity observation in the Limitations section rests on the measurements below. The Isharah pose set measured here spans 18 signers with 500 to 2,000 clips each, so its

Strategy	Clips	Types	Gloss cov.	H_{src}	Redund.	Tokens/type	Types $\geq$5
Random	702	476	0.439	0.846	0.448	11.73	181
Source-strat.	705	486	0.448	0.986	0.460	11.35	189
k-center	1491	440	0.406	0.831	0.530	13.90	178
Representativeness	1427	451	0.416	0.835	0.527	13.37	188
Articulatory only	827	497	0.458	0.837	0.453	11.33	208
Full obj. (art+repr+src)	1224	491	0.453	0.932	0.517	11.98	192
Text facility loc.	*1280*	*505*	*0.465*	*0.835*	*0.524*	*11.74*	*186*
Gloss oracle	*773*	*1024*	*0.944*	*0.853*	*0.485*	*6.06*	*276*

Table 10: Properties of the purchased subset at the 10 percent PHOENIX14T budget. H_{src} is normalized entropy over recording sources; redundancy is mean nearest-neighbor cosine similarity within the selected set. Tokens/type and Types $\geq$5 measure repetition within the purchase itself, not coverage of the pool. Gloss coverage is measured after selection and was not available to any label-free selector. Entropy sampling and BADGE are omitted: their purchases depend on a seed model that was not retained, so they cannot be reproduced exactly. Articulatory only and Full obj. were re-run for this table; both shifted by $\leq$2% from the originally reported run.

	40 codewords		150 codewords		400 codewords	
Budget	Random	Gain	Random	Gain	Random	Gain
0.5 h	1.000	0.000	0.997	0.003	0.847	0.143
1 h	1.000	0.000	1.000	0.000	0.942	0.055
2 h	1.000	0.000	1.000	0.000	0.981	0.019
5 h	1.000	0.000	1.000	0.000	0.997	0.003
10 h	1.000	0.000	1.000	0.000	0.999	0.001
40 h	1.000	0.000	1.000	0.000	1.000	0.000

Pose-corruption stability: duration-weighted overlap with clean selection

Strategy	10% dropout	20% dropout
Articulatory only	0.622	0.559
Full obj. (art+repr+src)	0.746	0.693
Representativeness	0.822	0.777

Table 11: Supporting diagnostics for articulatory coverage. The upper panel reports the fraction of the hand-configuration inventory covered by uniform random selection and the additional coverage obtained by greedy coverage selection. The 40-, 150-, and 400-codeword settings are diagnostic sweeps; the main selector uses 64 codewords. The lower panel reports duration-weighted overlap between selections made from clean and corrupted pose, where higher values indicate greater stability.

Test	PHOENIX14T	Isharah	Threshold
Signer leakage (excess NMI)	0.018	0.037	< 0.05
Temporal coherence (lift)	8.97$\times$	10.46$\times$	$> 3\times$
Channel independence (NMI)	0.047	0.048	< 0.30
Synthetic calibration, excess NMI			
handshape $\perp$ signer		0.007	
handshape $=$ signer		0.698	

Table 12: Validation of the articulatory channels. Both corpora sit near the independent control and far from the signer-determined one. This establishes that the channels are not degenerate; it does not establish that they correspond to linguistic categories.

Per-frame detection rate	PHOENIX14T *studio*	Isharah *smartphone*
Dominant hand, mean	0.771	0.836
Non-dominant hand, mean	0.720	0.712
Face	0.876	0.963
Dominant hand, worst signer	0.578	0.608
Dominant hand, best signer	0.834	0.974
Spread across signers	0.256	0.365

Table 13: Landmark detection rates. Upsampling PHOENIX14T frames by 2$\times$, 3$\times$ and 4$\times$ leaves detection essentially unchanged (0.691 to 0.687), so the limit is detail absent from the source rather than resolution. Smartphone footage recorded in unconstrained environments detects better than studio broadcast.

spread is not a small-sample artifact; the signer-independent training pool draws on 13 of them. On PHOENIX14T the extremes are the two smallest signers; restricted to signers with more than 500 clips the spread is 0.107.

Correlating per-signer detection rate with per-signer BLEU across all runs gives Spearman $\rho = 0.42$, $p = 0.27$ over nine signers. We report the input disparity, which is directly measured, and do not claim it causes worse translation.

G Implementation Details

Pose and normalization. Landmarks are extracted with MediaPipe Holistic (Lugaresi et al., 2019). Each frame is translated to the shoulder midpoint, scaled by shoulder width, and rotated so the shoulder axis is horizontal, so that body roll does not produce different location codewords for the same articulation. Coordinates are clipped to $[-5, 5]$ and median filtered with window five; gaps of at most five frames are linearly interpolated and longer gaps remain masked.

Articulatory channels. Six groups with codebook sizes 64 (hand configuration), 16 (orientation), 24 (location), 32 (motion), 8 (coordination) and 16 (nonmanual). Each hand is translated to its wrist, scaled by palm size, rotation aligned, and left hands are mirrored into the right-hand coordinate frame so the same configuration on either hand maps to one codeword. For handshape, orientation and location only frames below the clip's 60th speed percentile are used, since those parameters describe held configurations rather than transitions between them. Codebooks are fitted with MiniBatch k-means on descriptors standardized by median and interquartile range, sampling at most 200 descriptors per clip and 200,000 across the pool. Frames in which a hand is not detected are excluded before quantization; including them creates a codeword meaning "hand not found", which a coverage objective then treats as a rare pattern worth purchasing.

Pose quality is $q_i = \mathrm{clip}\big(\frac{v_i+a_i+r_i}{3}, 0.1, 1.0\big)$, where v_i is the fraction of frames with a detected hand, a_i is mean landmark availability across all tracked streams, and r_i is temporal continuity (the fraction of adjacent frames whose hand-validity does not flip); the floor at 0.1 down-weights poorly tracked clips without excluding them. The rarity weight is $r_{gk} = \min\Big(5, \sqrt{\frac{\mathrm{med}_g}{f_{gk}}}\Big)$, where f_{gk} is the pool frequency of codeword k in channel g and med_g is the median codeword frequency within that channel.

Downstream models. Gloss recognition uses a 4.1M-parameter model: frame projection to 256 dimensions, two temporal convolution blocks (kernel 5, stride 2), four pre-norm Transformer encoder layers with four heads and feed-forward width 1024, and a CTC output. Sequences longer than 320 frames are resampled. Translation uses a matched encoder with an autoregressive text decoder. Training uses AdamW at 3×10^{-4}, weight decay 10^{-2}, 5 percent warmup, cosine decay, at most 60 epochs with early stopping at patience 8, and an effective batch of 32.

Cost model. The pipeline supports two annotation-time multipliers, $6\times$ for translation and $40\times$ for gloss, the latter anchored on the How2Sign corpus's reported figure of approximately one annotator-hour per 90 seconds of video for gloss annotation (Duarte et al., 2021). Every budget reported in this paper, for both recognition and translation, uses the $6\times$ translation multiplier. Recognition budgets (2.8/5.5/11.0 h for PHOENIX14T, 7.1/14.2/28.4 h for Isharah) are 5/10/20 percent of each corpus's pool cost under that rate; Isharah translation reuses the same schedule. PHOENIX14T translation instead uses a finer 0.5/1/2/5-hour sweep, 0.9/1.8/3.6/9.1 percent of its pool cost. We use a common duration-based cost axis rather than task-specific annotation rates so that selector comparisons are not confounded by a change of cost model across tasks; this is a controlled-study convention, not a claim about the true relative annotation cost of glossing and translation. The $40\times$ gloss rate is not used to price any of the reported experiments. Budgets are expressed as the same fraction of pool annotation cost on both corpora rather than the same absolute hours, since Isharah's pool is 2.6 times larger.

Why the multiplier cannot affect the reported results. Pricing gloss recognition at the translation rate understates the true annotation cost, but it does not affect the selected subsets or downstream results. Let the multiplier be m, so clip i has cost $c_i = m\, d_i$, where d_i is its duration, and let the annotation budget be a fixed fraction β of the total pool cost, $B = \beta m \sum_j d_j$. The feasibility constraint $\sum_{i\in\mathcal{S}} m\, d_i \le \beta m \sum_j d_j$ then reduces to $\sum_{i\in\mathcal{S}} d_i \le \beta \sum_j d_j$, so the feasible subsets are independent of m. The greedy selection rule is invariant for the same reason: ranking candidates by marginal gain per purchased hour, $\Delta F/(m d_i)$, gives the same ordering as $\Delta F/d_i$ for any $m > 0$. Entropy sampling and BADGE are also unchanged because their seed sets are priced as a fixed 2 percent of the same target budget. Replacing the $6\times$ rate with the $40\times$ gloss rate therefore leaves every purchased subset and downstream result unchanged; it only rescales the reported annotation hours by $40/6$. The corresponding recognition budgets would be 18.7/36.7/73.3 h for PHOENIX14T and 47.3/94.7/189.3 h for Isharah. This invariance follows directly from the budget definition and selection rule rather than from an empirical sensi-

tivity analysis. The rate assumption still affects comparisons of recognition and translation budgets in absolute annotator hours, since the two tasks should not be interpreted as having the same real annotation cost.

Label isolation. The constraint that selection never sees a label is enforced by a static analysis test that parses each selection module and fails if it reads a gloss or translation field, by attribute or by subscript. Naming an annotation type in the cost model is permitted, since the cost model must know the relative price of a gloss and a translation.

Shared-Encoder Contrastive Grounding for Pose-Based Indian Sign Language Translation

Gyanendra Shrestha
Florida State University
gshrestha@fsu.edu

Liting Zhang
University of West Florida
lzhang@uwf.edu

Abstract

The WSLP 2026 shared task studies sentence-level translation from Indian Sign Language pose sequences to English, with training on iSign and evaluation on a disjoint collection of source videos. We address two challenges in this setting: unstable pose scale caused by landmark errors and weak grounding between pose inputs and generated text. We introduce robust clip-level normalization and shared-encoder contrastive grounding, which aligns pose and target-text representations in a pretrained T5 encoder during training while retaining standard autoregressive decoding at inference. On a video-held-out iSign development split, contrastive grounding improves chrF from 0.102 to 0.121, while clip-level normalization further improves it to 0.128; the normalization gain is consistent across three matched seeds. A retrieval probe provides complementary evidence of improved pose–text alignment. Our final shared-task submission achieves chrF 0.16 on the official test set. However, input-independent English controls reach chrF 0.15, showing that small differences in the official score are difficult to interpret at the current level of translation accuracy. The source code is available at https://github.com/gyan-shrestha/WSLP-2026-SLT.

1 Introduction

The WSLP 2026 shared task studies sentence-level translation from Indian Sign Language (ISL) pose sequences to English. Training supervision comes from the public iSign corpus (Joshi et al., 2024), while the shared-task validation and test clips are drawn from a separate set of source videos with no overlap with iSign. Models must therefore transfer from iSign to a different collection of recordings and pose estimates without labeled examples from the evaluation collection.

Two problems became apparent in this setting. First, a pretrained encoder–decoder can produce fluent English without learning a strong connection between the pose sequence and the sentence content. In preliminary experiments, unrelated clips often received the same generic predictions, indicating that the language model was easier to learn than the pose-to-text mapping. Second, frame-wise pose normalization was sensitive to unreliable shoulder estimates. Near-zero shoulder widths occasionally produced extreme coordinate values, and these failures were more common in the shared-task clips than in iSign.

We address these problems with robust clip-level normalization and shared-encoder contrastive grounding. Pose sequences are projected into the embedding space of a pretrained T5 encoder–decoder. During training, the corresponding target sentence is encoded by the same T5 encoder, and a contrastive objective aligns the pooled pose and text representations alongside the translation loss. The text branch is removed at inference. For pose preprocessing, scale is estimated from the median shoulder width over the clip rather than independently in each frame, reducing the effect of individual landmark failures.

Controlled experiments on held-out iSign videos show that contrastive grounding provides the largest improvement over cross-entropy training, with clip-level normalization providing a further gain. Repeating the normalization comparison across three matched seeds yields higher chrF with normalization in all three runs. A retrieval-based grounding probe provides complementary evidence that the contrastive objective affects pose–text alignment directly: the cross-entropy-only checkpoint performs near chance, while the contrastive checkpoints achieve substantially higher retrieval recall. Several other changes, including velocity features, augmentation, a larger T5 model, and masked-pose pretraining, do not show a similarly clear improvement.

The shared-task evaluation also exposes a limi-

Proceedings of the Second Workshop on Sign Language Processing, pages 132–140
October 29, 2026 ©2026 Association for Computational Linguistics

tation of interpreting the official score in isolation. The same trained checkpoint receives different chrF scores under different output-length constraints, while input-independent English can recover a substantial portion of the score. We therefore report the controlled method evaluation separately from the analysis of the official test output.

The main contributions are:

- An analysis of the WSLP evaluation showing that, at the current level of translation accuracy, chrF is strongly affected by decoding choices and English surface form in addition to the signed input. A fixed English sentence selected using only the output register, without using pose, scores within 0.02 chrF of the top official submission (Section 4.6).
- A pose-to-text formulation that jointly trains autoregressive translation and pose–text contrastive alignment through the same pretrained T5 encoder, without requiring a separate text encoder at inference.
- A clip-level normalization procedure that reduces the effect of unstable pose scale across recording collections and provides an additional improvement over contrastive training alone.

The model combines established components: pose projection into a pretrained T5 encoder–decoder (Uthus et al., 2023), contrastive visual–text alignment (Zhou et al., 2023), and an InfoNCE objective (van den Oord et al., 2018; Radford et al., 2021). Their combination is computationally inexpensive, adds no inference-time parameters, and allows the alignment objective to act directly on the encoder states used by the translation decoder.

2 Related Work

Pose-to-text translation. The iSign benchmark introduced a large ISL–English corpus together with pose-to-text baselines for open-domain translation, with its strongest reported pose baseline reaching BLEU-4 1.47 on the conventional 0–100 scale (Joshi et al., 2024). A related pose-to-text formulation projects pose features into the embedding space of a pretrained T5 model and trains the encoder–decoder directly for translation (Uthus et al., 2023). Our backbone follows this general design, while the WSLP task introduces a different evaluation setting in which the test clips come from a disjoint collection and no labeled target-domain development set is available.

Visual-language alignment. Early neural SLT systems were developed on constrained corpora such as PHOENIX14T (Camgoz et al., 2018), where jointly modeling recognition and translation is effective (Camgoz et al., 2020). Gloss-free SLT work has used visual-language pretraining to strengthen the connection between visual input and target text. GFSLT-VLP (Zhou et al., 2023), for example, learns aligned visual and textual representations before translation. We use the same broad contrastive principle, but place it inside end-to-end translation training: pose and target text share the T5 encoder, and the alignment loss regularizes the encoder states attended to by the decoder.

Motion and ISL modeling. Motion-aware modeling has shown stronger performance on the iSign test split, with BLEU-4 reaching 8.58 when optical flow is combined with multi-scale spatial encoding and a language-model decoder (Chowdhury and Sanyal, 2025). Although this setting differs from WSLP in both input representation and evaluation data, the result indicates that video-derived motion can provide information that sparse pose coordinates do not capture. For isolated ISL recognition, MediaPipe keypoints combined with a temporal convolutional network achieve 54% top-1 accuracy across 4,361 classes (Reddy and Kamakshi, 2025). Our work considers a different setting: sentence-level, open-domain translation from the pose sequences provided for WSLP Task 1.

3 Methodology

The proposed model translates ISL pose sequences directly into English using a pretrained T5 encoder–decoder. Figure 1 shows the overall pipeline. A temporal frontend first converts the pose sequence into T5-compatible embeddings. These embeddings are processed by the T5 encoder and passed to the decoder for autoregressive translation. During training, the target sentence is also encoded by the same T5 encoder, and a contrastive objective encourages paired pose and text representations to remain close while separating other sentences in the batch. The final training objective combines translation and contrastive losses:

$$\mathcal{L} = \mathcal{L}_{\mathrm{MT}} + \lambda_{\mathrm{con}} \mathcal{L}_{\mathrm{con}} \tag{1}$$

where $\mathcal{L}_{\mathrm{MT}}$ is token-level cross-entropy and

$\lambda_{\text{con}} = 0.5$. The target-text branch is used only during training. Register conditioning and multi-seed ensemble decoding complete the full system.

3.1 Pose representation and normalization

The released pose features contain 576 MediaPipe Holistic (Lugaresi et al., 2019) landmarks per frame at 25 fps. We retain 103 landmarks covering both hands, the upper-body joints, and 40 facial points around the lips, eyes, and brows. Each frame is represented by the corresponding two-dimensional coordinates.

Pose scale varies with camera distance and with the quality of landmark detection. A frame-wise normalization based on shoulder width is particularly unstable when one shoulder is missing or poorly localized, since a very small estimated width can produce extremely large coordinates. To reduce this sensitivity, each frame is centered on its shoulder midpoint, while scale is estimated from the median shoulder width over the full clip. The sequence is then standardized to zero mean and unit variance. This retains relative articulatory motion while reducing large frame-to-frame changes caused by pose-estimation errors.

3.2 Pose-to-text architecture

The temporal frontend contains a linear projection followed by two stride-2 convolutions, LayerNorm, and a learned scaling parameter. The two convolutions reduce the temporal resolution by a factor of four and give each output position a receptive field of approximately nine input frames. The resulting sequence is projected to the dimensionality of `flan-t5-base` (Raffel et al., 2020; Chung et al., 2024) and supplied through `inputs_embeds`. T5 then performs sequence encoding and autoregressive generation in the same way as for ordinary token embeddings.

Direct pose embeddings also need to match the scale of the representations seen during T5 pretraining. In the checkpoint used here, normal token embeddings have a mean norm near 300, whereas the LayerNorm output from the pose frontend has norm $\sqrt{d_{\text{model}}} = 27.7$. Feeding the latter without rescaling produced an initial loss near 609. A learned scalar is therefore initialized from the measured mean norm of the checkpoint embedding table before joint training.

Split	Clips	Videos	References
iSign, raw	124,194	3,726	yes
Task validation	5,256	1,804	no
Task test	5,333	1,813	no
Video-id overlap, iSign ∩ evaluation: **0**			

Table 1: Data used for WSLP Task 1. The evaluation clips come from source videos that do not occur in iSign.

3.3 Shared-encoder contrastive grounding

Cross-entropy alone optimizes the probability of the target sentence but does not directly require the pose representation to distinguish one sentence from another. In preliminary runs, this often led to similar generic English predictions for unrelated pose sequences. The contrastive objective adds sentence-level discrimination to the same encoder used for translation.

Let H_i^p and H_i^t denote the shared T5 encoder states for pose sequence x_i and target sentence y_i. Mean pooling followed by ℓ_2 normalization gives z_i^p and z_i^t. For a batch of size B, the contrastive loss is

$$\mathcal{L}_{\text{con}} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\exp(\text{sim}(z_i^p, z_i^t)/\tau)}{\sum_{j=1}^{B}\exp(\text{sim}(z_i^p, z_j^t)/\tau)} \tag{2}$$

where sim is cosine similarity and $\tau = 0.07$. This is the InfoNCE objective of (van den Oord et al., 2018), applied across modalities as in (Radford et al., 2021). Other target sentences in the batch serve as negatives. Since the two branches share the T5 encoder, the additional objective acts on the representation used by the translation decoder and introduces no separate text encoder at inference.

3.4 Dataset and preprocessing

Training uses the public iSign corpus (Joshi et al., 2024), while validation and test evaluation follow WSLP 2026 Task 1. The raw iSign file contains 124,194 sentence-level pairs from 3,726 source videos. The WSLP release contains 5,256 validation clips from 1,804 videos and 5,333 test clips from 1,813 videos. Comparing the source-video identifiers shows no overlap between iSign and the shared-task evaluation videos.

The WSLP validation release does not contain reference translations, so development experiments use an iSign holdout. A total of 1,940 clips are

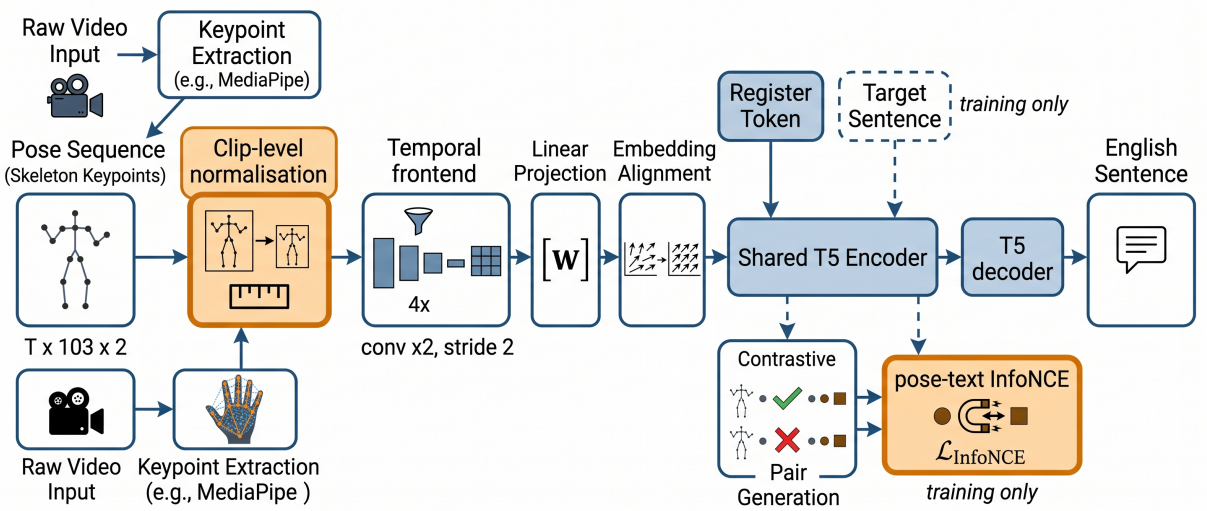


Figure 1: Overview of the proposed pose-to-text translation model. Orange boxes are the two changes introduced in this work: clip-level pose normalization and shared-encoder contrastive grounding. The pose sequence is normalized, reduced in time by a convolutional frontend, and projected into the embedding space of a pretrained T5 encoder–decoder together with a register control token. During training the target sentence is encoded by the *same* T5 encoder and an InfoNCE objective aligns the pooled pose and text representations, so the contrastive term regularizes exactly the states the decoder attends to and adds no parameters at inference. Dashed boxes and arrows are used only during training and are removed at inference.

held out by source video rather than by clip, avoiding nearby segments from the same recording on both sides of the split. The resulting development set remains in-domain with respect to iSign and is therefore used for controlled model comparison rather than as a direct estimate of WSLP test performance.

Before splitting the data, 3,408 iSign targets consisting of a single token, bare digits, or punctuation-like text are removed, leaving 120,786 usable pairs. The most frequent target in the raw file is 2. These examples are excluded because they strongly favor degenerate short predictions.

Two uid patterns in the evaluation data correspond to different target styles: capitalized, punctuated prose and lowercase fragments. Since the uid identifies this distinction at inference time, a control token is prepended to the pose embeddings to specify the target register. Training labels for this token are derived from the target surface form.

3.5 Implementation details

Training uses AdamW with separate learning rates for the randomly initialized pose frontend and the pretrained T5 backbone. The frontend uses 1×10^{-4} and T5 uses 3×10^{-5}, with cosine scheduling and 8% linear warmup. A single learning rate of 1×10^{-4} for both components caused the loss to rise from 5.8 to 15.1 near the peak of the schedule.

Decoder	`flan-t5-base` (223M)
Pose frontend	6.1M parameters
Input	103 keypoints $\times 2$, 25 fps
Max input / target	512 frames / 48 tokens
Optimizer	AdamW, $\beta = (0.9, 0.98)$
LR, frontend / T5	10^{-4} / 3×10^{-5}
Schedule	cosine, 8% linear warmup
Weight decay / clipping	0.01 / 1.0
Batch size / epochs	64 / 14
Precision	bfloat16
Contrastive weight	0.5
Contrastive temperature	0.07
Development set	1,940 clips, held out by video
Hardware / runtime	1 × B200 / ≈60 min per model
Ensemble	5 models (chrF 0.15)
Submitted output	single model + fixed suffix
Submitted decoding	beam 8, length penalty 2.0
	no-repeat 3-grams, min. 36 tokens

Table 2: Training and decoding configuration.

Models are trained for 14 epochs with batch size 64, weight decay 0.01, gradient clipping at 1.0, and bfloat16 precision. The contrastive weight is 0.5 and the temperature is 0.07.

The submitted model averages token log-probabilities from five runs that differ only in random seed and data order. Decoding uses beam size 8, length penalty 2.0, no-repeat 3-grams, and a minimum output length of 36 tokens. The local evaluation follows the shared-task metric stack: unsmoothed BLEU (Papineni et al., 2002) with the NLTK punkt tokenizer, ROUGE (Lin, 2004)

Configuration	chrF	ROUGE-L	BLEU
Cross-entropy only	0.102	0.076	0.0089
+ shared-encoder contrastive grounding	0.121	0.089	0.0113
+ robust clip-level normalization	**0.128**	**0.094**	**0.0115**

Table 3: Ablation results on the video-held-out iSign development split.

via `rouge==1.0.1`, and chrF (Popović, 2015) via sacreBLEU (Post, 2018). As a scorer check, a constant `hello` prediction receives chrF 0.0317 locally and 0.03 from the official scorer.

4 Results

4.1 Shared-task performance

On the official WSLP 2026 Task 1 test set, our submitted system reaches chrF 0.16, ROUGE 0.08, and BLEU 0.00, ranking second among five submitted systems. Official chrF scores range from 0.13 to 0.17, while all five systems obtain BLEU at or below 0.02. Under the same decoding constraints, both the five-model ensemble and the single strongest model reach chrF 0.15; the remaining 0.01 comes from a fixed suffix appended to every prediction, analyzed in Section 4.5. Ensembling therefore improves the development scores reported below without producing a measurable gain on the evaluation collection. Because the official test references are not released, the effect of individual model components is evaluated on the video-held-out iSign development split.

4.2 Ablation study

Because the official test references are not released, component-level comparisons use the video-held-out iSign development split. Section 4.4 shows that performance on this split does not always predict official chrF, so the ablation results characterize the method on held-out iSign videos rather than establish the same component ordering on the WSLP collection.

The controlled development experiments in Table 3 isolate the two main changes in the proposed method. Cross-entropy training alone reaches chrF 0.102, ROUGE-L 0.076, and BLEU 0.0089. Adding shared-encoder contrastive grounding increases all three metrics and raises chrF to 0.121. Replacing frame-wise scaling with clip-level normalization further raises chrF to 0.128.

The normalization change also addresses a measurable difference in input scale. Under frame-wise shoulder scaling, training features have standard deviation 1.99 and a 99.9th percentile of 3.29, while the corresponding test statistics are 3.04 and 13.92. After clip-level standardization, both collections have unit variance. Together with the ablation result, this indicates that clip-level normalization reduces a measurable scale mismatch between the two collections.

Seed	chrF	ROUGE-L	BLEU
0	0.1240	0.0953	0.0142
1	0.1305	0.0849	0.0090
2	0.1301	0.1024	0.0113
mean ± sd	0.128 ± 0.003	0.094 ± 0.007	0.0115 ± 0.0021
range	0.0065	0.0175	0.0052

Table 4: Three runs of the final configuration, with contrastive grounding and clip-level normalization, using different random seeds.

Seed	chrF	ROUGE-L	BLEU
0	0.1212	0.0893	0.0113
1	0.1166	0.0952	0.0123
2	0.1024	0.0911	0.0071
mean ± sd	0.113 ± 0.008	0.092 ± 0.003	0.0102 ± 0.0022
range	0.0187	0.0059	0.0052

Table 5: Three runs of the contrastive configuration *without* normalization, same seeds as Table 4.

The contrastive objective also learns a measurable pairing signal. Its loss decreases from 4.45 to 2.66, compared with $\ln 64 = 4.16$ at the batch size used here. This shows that the shared encoder learns to distinguish the paired sentence from the other sentences in its batch, but batch-level discrimination does not by itself establish semantic grounding. The rank-based retrieval probe in Section 4.6 provides a more direct test: the correct sentence ranks substantially above chance for the contrastive checkpoints and near chance for the cross-entropy-only checkpoint. Translation performance remains the primary evidence for the method; the retrieval probe provides complementary evidence of alignment beyond batch discrimination.

4.3 Run-to-run variation

Table 4 reports three runs of the final configuration, which combines contrastive grounding with clip-level normalization. chrF ranges from 0.1240 to 0.1305, with a mean of 0.128 ± 0.003.

To assess the normalization gain across random seeds, we repeated the contrastive configuration without clip-level normalization using the same three seeds as the final configuration (Table 5).

Normalization improves chrF in all three

Configuration	chrF	ROUGE-L	BLEU
Contrastive (reference)	0.121	0.089	0.0113
+ velocity features	0.121	0.081	0.0099
+ pose augmentation	0.116	0.083	0.0095
+ `flan-t5-large`	0.106	0.074	0.0083
+ masked-pose pretraining	0.118	0.090	0.0078
+ all of the above	0.110	0.079	0.0084

Table 6: Single-run modifications on the video-held-out iSign development split, each varying one component from the contrastive configuration.

matched comparisons, by $+0.0028$, $+0.0139$, and $+0.0277$, with an average improvement of $+0.015$. The magnitude varies across runs, but the direction is consistent across all three seeds. ROUGE-L and BLEU vary more across runs, so small differences on those metrics should be interpreted cautiously.

Other single-run modifications, including velocity features, pose augmentation, `flan-t5-large`, and masked-pose pretraining, do not improve beyond the variation observed here. Table 6 reports each, varying one component from the contrastive (no-normalization) configuration; none of these differences is larger than that configuration's own seed range (0.0187, Table 5), so we report them as unresolved rather than as evidence that the component is harmful.

We discuss two of these results in more detail below.

Velocity features. Appending first-order frame differences to the keypoint stream did not help. This differs from prior work showing substantial gains from optical flow computed directly on video (Chowdhury and Sanyal, 2025). First differences of 103 keypoint coordinates provide a much coarser motion signal than dense optical flow, so this negative result should not be interpreted as evidence against motion representations more broadly.

Masked-pose pretraining. We pretrained the frontend by masked reconstruction over iSign together with the unlabeled WSLP validation and test poses, as a form of unsupervised adaptation to the evaluation collection. It reduced development loss while producing lower translation scores. Reconstructing masked pose frames rewards features that predict neighboring coordinates, which is not the same as features that carry sentence-level content.

4.4 Cross-collection development

Performance on the iSign development split does not consistently predict performance on the WSLP collection. Across three submitted configurations, development BLEU rises from 0.0073 to 0.0188, whereas official chrF changes from 0.12 to 0.13 and then falls to 0.11. The configuration with the strongest development result is therefore the weakest of the three under official chrF.

This difference is consistent with the collection split described in Section 3: the development set contains held-out iSign videos, while the shared-task clips come from a separate set of source videos. Holding out complete videos reduces within-recording leakage, but it does not reproduce the target distribution.

4.5 Effect of decoding on the official score

The ablation above measures changes to the trained model, while the final WSLP score also depends on how that model is decoded. Table 7 evaluates one fixed checkpoint under six different output-length constraints. Increasing the mean output length from 11.6 to 25.9 words raises chrF from 0.13 to 0.15 even though the model itself is unchanged, and lengthening further to 60.9 words brings it back down to 0.14. Over the same sweep ROUGE-L decreases monotonically from 0.09 to 0.06 and BLEU remains 0.00, so the two metrics move in opposite directions for a model that never changed. For scale, that 0.02 swing from decoding alone is half the 0.04 spread separating all five submitted systems.

The final row of Table 7 appends the same English sentence to every 25.9-word prediction. This text is identical for all 5,333 clips and does not use the pose, yet it increases chrF from 0.15 to 0.16. The final 0.01 in the submitted score therefore comes from decoding rather than a change in the learned translation model.

4.6 Input-independent controls

The decoding results suggest that part of the chrF score may come from target-language surface form rather than the signed input. To estimate this effect directly, Table 8 evaluates several controls that do not use the pose sequence.

A fixed sentence repeated for every test clip reaches chrF 0.14, while selecting a fixed sentence using only the known output register raises the score to 0.15. The random-sentence control also

Mean output length	chrF	ROUGE	BLEU
11.6 words	0.13	0.09	0.00
19.6 words	0.14	0.09	0.00
25.9 words	**0.15**	0.08	0.00
32.6 words	**0.15**	0.08	0.00
42.1 words	**0.15**	0.07	0.00
60.9 words	0.14	0.06	0.00
25.9 words + fixed suffix (43.9)	**0.16**	0.08	0.00

Table 7: Official test scores for one fixed checkpoint under different output-length constraints. The final row appends the same fixed suffix to every prediction.

Prediction rule	chrF	ROUGE	BLEU
constant `hello`	0.03	0.00	0.00
one fixed sentence	0.14	0.09	0.00
fixed sentence, register-matched	**0.15**	0.08	0.00
random training sentence	0.13	0.04	0.00

Table 8: Sanity checks on the official test set. None of these controls use the pose sequence.

reaches 0.13. None of these predictions uses information from the signed content. All four controls were evaluated with the official scorer and are therefore directly comparable with the leaderboard scores in Section 4. Three fall within the 0.13–0.17 chrF range spanned by the five submitted systems, and the best input-independent control is only 0.02 below the top submission. In contrast, the constant `hello` control reaches only 0.03 chrF, indicating that the higher scores depend on plausible English surface form rather than arbitrary output.

Because the contrastive objective aligns pose and text in a shared space, the trained encoder can also be used as a retrieval index, giving a pose-conditioned control that the controls above lack. We embedded all training clips, embedded each development clip, and returned the nearest training sentence. On the development split, this reaches chrF 0.154, ROUGE-L 0.067, and BLEU 0.0069, compared with chrF 0.121 for generated output from the same checkpoint on the same split. We do not report this result on the official test set because no target-domain retrieval index is available for the WSLP evaluation collection. Retrieved sentences are fluent and human-written but, on inspection, unrelated in meaning to the query clip: the retrieval control separates two properties that chrF does not distinguish, since a retrieved sentence has ideal surface fluency and essentially no content accuracy, and still scores above generated output that is partially correct.

Checkpoint	R@1	R@5	R@10
Cross-entropy only	0.0005	0.0026	0.0057
+ contrastive	0.0072	0.0294	0.0567
+ normalization	**0.0351**	**0.1191**	**0.1964**
chance (N=1,940)	0.0005	0.0026	0.0052

Table 9: Pose-to-text retrieval recall on the 1,940-clip development split. R@1/5/10 reports the fraction of pose queries for which the paired reference sentence appears among the top 1, 5, or 10 candidates. Chance recall is shown for reference.

Retrieval recall as a grounding probe. The contrastive loss measures discrimination within a training batch, but does not show how well pose and text representations align on held-out examples. We therefore evaluate pose-to-text retrieval on the 1,940-clip development split. For each clip, we encode its pose and reference sentence separately with the shared T5 encoder, mean-pool the representations, and rank all 1,940 reference sentences by cosine similarity to the pose query. We report recall at 1, 5, and 10 (R@1/5/10), together with chance performance for a candidate pool of this size. The cross-entropy-only checkpoint provides a control because it was not trained with the contrastive objective.

The cross-entropy-only checkpoint performs near chance at each k, while both contrastive checkpoints perform substantially above chance. Contrastive grounding reaches R@10 of 0.057, compared with a chance level of 0.0052, and adding normalization raises R@10 to 0.196. The median rank of the correct sentence also improves from 158 to 49 out of 1,940 candidates after normalization. These results provide additional evidence that contrastive training produces pose–text alignment beyond the decrease in its training loss, while normalization further improves retrieval on this probe. Absolute retrieval performance nevertheless remains limited: at R@10 of 0.196, the correct sentence falls outside the top 10 for about four fifths of the queries.

Taken together, these results show that substantial character-level overlap can be obtained from English surface form alone in this evaluation setting, whether or not the prediction is conditioned on the pose. They do not imply that chrF is generally unsuitable for sign language translation; rather, they show that chrF is difficult to interpret on its own when translation accuracy is this low.

Reference: *Do you know whose idea it was to set up Clin d'Oeil? It was him!*	
CE only	This is why they have a lot of people.
+ contrastive	It is a very important importance of the world.
Reference: *Hello! Hello! Hello!*	
CE only	This is why you will be very happy.
+ contrastive	Let me tell you.
Reference: *It is all about Sign Language. Which is very important.*	
CE only	This is why it was very happy.
+ contrastive	This is why he was very happy.

Table 10: Development examples from cross-entropy and contrastive training.

4.7 Qualitative analysis

The generated sentences are often fluent even when their content is poor. Under cross-entropy alone, unrelated development clips frequently receive similar sentences beginning with "This is why". Contrastive training reduces this repetitive pattern and increases output variation, although the examples in Table 10 remain far from their references. Before decoding calibration, one sentence accounts for 9.2% of the 5,333 official test predictions and only 23% of predictions are distinct.

The qualitative results are consistent with the quantitative ablation but also limit the strength of the grounding claim from generation alone. Shared-encoder contrastive training improves the measured translation scores and reduces an obvious repetition mode, but the generated sentences remain weakly aligned with the reference content; the retrieval probe in Section 4.6 is what supports a grounding claim beyond generation quality, and even there the absolute recall remains low.

Taken together, these experiments separate two sources of improvement. The controlled ablation shows that contrastive grounding and clip-level normalization improve the trained model on the available development data. The decoding and input-independent controls show that changes of a similar magnitude in official chrF can also arise without changing the model itself. This makes small chrF differences difficult to interpret on their own in the current low-accuracy setting.

5 Conclusion

This paper presents a pose-to-text approach for the WSLP 2026 Indian Sign Language translation task. The method combines robust clip-level pose normalization with shared-encoder contrastive grounding, using the same T5 encoder for pose and target text during training. Controlled experiments on held-out iSign videos show improved translation scores after contrastive grounding and clip-level normalization. Across three matched seeds, normalization improves chrF in every comparison. A rank-based retrieval probe provides complementary evidence that contrastive training improves pose–text alignment, with a further improvement after normalization.

The shared-task evaluation reveals a separate difficulty. Performance on held-out iSign does not consistently predict performance on the WSLP collection, and the official chrF score is sensitive to output length and input-independent English content. These results support the proposed training method while also showing that, at the current level of translation accuracy, small changes in chrF do not necessarily correspond to better use of the signed input.

Limitations

The method is evaluated on a single sign language and a single training corpus (iSign); whether shared-encoder contrastive grounding transfers to other sign languages or pose-extraction pipelines is untested. Register conditioning relies on a uid pattern specific to the WSLP evaluation data, and would need a learned or predicted register signal in a setting where this side information is unavailable at inference. The contrastive objective improves translation scores and reduces a specific repetition failure mode, and a rank-based retrieval probe (Section 4.6) shows the resulting pose representation is above-chance predictive of its paired sentence, but the qualitative examples in Table 10 show generation itself does not establish accurate content grounding: generated sentences often remain unrelated to the reference in meaning even when fluent, and even the retrieval probe's absolute recall is low (R@10 of 0.20 on a 1,940-clip pool). Because official test references are not released for this shared task, all component-level comparisons rely on a held-out development split, which Section 4.4 shows can rank configurations differently than the official evaluation; the true ranking of the tested components on the WSLP test collection therefore remains uncertain. The retrieval probe is subject to the same limitation.

Absolute translation quality also remains low: our system obtains BLEU 0.00, and every submit-

ted system obtains BLEU at or below 0.02. At this level of performance, small chrF differences cannot by themselves establish better use of the signed input.

Acknowledgments

We thank anonymous reviewers for their valuable feedback.

References

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Anal Roy Chowdhury and Debarshi Kumar Sanyal. 2025. Enhancing indian sign language translation via motion-aware modeling. In *Proceedings of the Workshop on Sign Language Processing (WSLP)*, pages 39–46.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Abhinav Joshi, Romit Mohanty, Mounika Kanakanti, Andesha Mangla, Sudeep Choudhary, Monali Barbate, and Ashutosh Modi. 2024. isign: A benchmark for indian sign language processing. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10827–10844.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. MediaPipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Tatigunta Bhavi Teja Reddy and Vidhya Kamakshi. 2025. Pose-based temporal convolutional networks for isolated indian sign language word recognition. In *Proceedings of the Workshop on Sign Language Processing (WSLP)*, pages 47–50.

David Uthus, Garrett Tanzer, and Manfred Georg. 2023. YouTube-ASL: A large-scale, open-domain American sign language-English parallel corpus. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Benjia Zhou, Zhigang Chen, Albert Clapés, Jun Wan, Yanyan Liang, Sergio Escalera, Zhen Lei, and Du Zhang. 2023. Gloss-free sign language translation: Improving from visual-language pretraining. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

CARE: A Coverage-Aware Retrieval Ensemble for Isolated Sign Language Recognition

Nguyen Tu Nam and Hiroki Takahashi
The University of Electro-Communications, Tokyo, Japan
g2340024@gl.cc.uec.ac.jp, rocky@inf.uec.ac.jp

Abstract

Isolated sign language recognition is commonly formulated as supervised classification, but this formulation is challenging when nearly every target gloss has only one labeled video. We study the Isolated Sign Language Recognition track of the WSLP 2026 Shared Task on Indian Sign Language Processing, a predominantly one-shot target setting with 4,398 training videos spanning 4,361 glosses, over 99% of which are singletons. We introduce CARE, a Coverage-Aware Retrieval Ensemble that augments a complete local visual dictionary with a CISLR reference bank covering a subset of the target vocabulary. A frozen Video-Swin encoder ranks all target labels locally, while frozen I3D and SignRep encoders provide external retrieval evidence for lexically matched labels. CARE estimates orthogonal Procrustes mappings from shared-class centroids, retains both raw and aligned representation geometries, and preserves external videos as individual nearest-neighbor references. Query-wise score standardization and a deterministic coverage mask restrict external evidence to supported labels. The method requires neither target-specific backbone fine-tuning nor a large-vocabulary classification head. CARE obtains 74.31%/76.15%/76.15% Top-1/5/10 accuracy on the WSLP 2026 validation split and 65.59%/67.11%/68.25% on the official test split. The 8.72% Top-1 gap warrants caution when interpreting performance on the small 109-query validation split used for model selection.

1 Introduction

Isolated sign language recognition (ISLR) maps a sign video to its corresponding lexical label, or gloss. ISLR can support dictionary lookup and corpus annotation, but achieving robust recognition under unconstrained conditions remains a challenge. In this paper, we address the Isolated Sign Language Recognition track of the WSLP 2026 Shared Task on Indian Sign Language Processing.[1] We refer to the task's word-recognition dataset as WR.

WR presents a predominantly one-shot regime in which nearly all target glosses occur exactly once in the training set. A standard classifier would therefore have almost no within-class evidence for estimating most decision boundaries. An alternative formulation interprets the labeled target set as a visual dictionary: each training video acts as a reference entry, reducing recognition to a query-to-entry ranking problem. This retrieval formulation also permits vocabulary expansion without retraining a classification head, a property explored by prior work on one-shot and dictionary-based sign recognition (Joshi et al., 2022; De Coster and Dambre, 2023; Desai et al., 2023; Vandendriessche et al., 2025). CARE builds on this formulation by augmenting the complete target dictionary with a cross-dataset reference bank that supports only a known subset of labels. Its coverage-aware fusion prevents labels without external references from receiving external scores.

Public labeled corpora can enlarge the reference dictionary, but their use introduces three coupled problems. First, dataset-specific differences in signers, temporal boundaries, and recording conditions can shift feature distributions even when the same encoder is used. Second, the external lexicon only partially overlaps the target vocabulary. Normalized exact matching with the public CISLR corpus (Joshi et al., 2022) supports only part of the target vocabulary; all other labels must still compete in the same global ranking. Third, distances produced by different representations and geometries can have different distributions. Assigning large external distances to unsupported labels would conflate absence of external support with negative vi-

[1] https://exploration-lab.github.io/WSLP-2026/task/

Proceedings of the Second Workshop on Sign Language Processing, pages 141–146
October 29, 2026 ©2026 Association for Computational Linguistics

sual evidence.

We address this setting with CARE. A frozen Video-Swin (Liu et al., 2022) encoder supplies a complete local dictionary score, while frozen WLASL-adapted I3D (Carreira and Zisserman, 2017; Li et al., 2020) and SignRep (Wong et al., 2025) encoders provide external CISLR retrieval scores for the lexically covered labels. We estimate orthogonal Procrustes mappings from shared-gloss class centroids and retain both raw and aligned representations. The resulting scores are standardized per query and fused through a deterministic coverage mask, so unsupported labels retain only the local score. CARE thus combines local retrieval, cross-dataset alignment, and coverage-aware fusion without target-specific backbone fine-tuning.

2 Related Work

2.1 Large-Vocabulary and Low-Resource ISLR

Large-scale datasets have shifted ISLR from small controlled vocabularies toward recognition under substantial variation in signer and recording conditions. MS-ASL introduced a signer-independent benchmark with 1,000 American Sign Language labels (Vaezi Joze and Koller, 2019), while WLASL provides videos covering 2,000 word classes and compares appearance- and pose-based models (Li et al., 2020). These resources established strong supervised baselines and encouraged the use of spatio-temporal networks such as I3D (Carreira and Zisserman, 2017) and hierarchical video transformers such as Video Swin (Liu et al., 2022).

These benchmarks mostly support fixed classifiers trained with repeated examples per class. One approach to low-resource recognition transfers pretrained representations across datasets or languages. OpenHands standardizes pose-based representations across six sign languages (Selvaraj et al., 2022). Most relevantly, CISLR proposes prototype-based one-shot recognition for Indian Sign Language using features transferred from resource-richer American Sign Language data (Joshi et al., 2022). SignBERT adds a prediction head and fine-tunes its pretrained encoder for supervised isolated-sign classification (Hu et al., 2021). This setup differs from our WR setup, where nearly all target classes have only one labeled example and we formulate recognition as retrieval. SignRep, by contrast, demonstrates that a frozen encoder can support sign-dictionary retrieval (Wong et al., 2025). We therefore use it in CARE's external retrieval branch.

2.2 Dictionary Retrieval

Several works formulate sign recognition as query-by-example search. De Coster and Dambre retrieve dictionary entries with dense vector search (De Coster and Dambre, 2023), while Boháček and Hrúz study few-shot recognition using videos collected from online dictionaries (Boháček and Hrúz, 2023). ASL Citizen frames large-vocabulary ISLR as dictionary retrieval (Desai et al., 2023), and Vandendriessche et al. (Vandendriessche et al., 2025) learn representations for one-shot retrieval across changing dictionaries and sign languages. Metric-based few-shot learning provides a related comparison: Matching Networks predict from attention-weighted query–support similarities (Vinyals et al., 2016), whereas Prototypical Networks compare each query with the mean embedding of a class's support examples (Snell et al., 2017). CARE is closer to instance-level matching: although we use shared-class centroids to estimate the alignment maps, retrieval retains every external video rather than replacing each class with a single prototype. It additionally combines a complete local dictionary with an external bank that covers only a known subset of target labels.

2.3 Cross-Dataset Alignment and Partial Coverage

Cross-dataset retrieval must account for both representation shift and incomplete label overlap. CORAL aligns second-order feature statistics (Sun et al., 2016), and Procrustes analysis has been used for manifold alignment (Wang and Mahadevan, 2008). Universal domain adaptation considers shared and private source and target classes (You et al., 2019), but must typically infer their relationship. In CARE, normalized lexical matching directly identifies the externally supported target labels. We estimate an orthogonal map from shared-gloss centroids, retain the raw geometry as a complementary channel, and mask external scores for unsupported labels.

3 Methodology

3.1 Problem Formulation and Overview

Let $\mathcal{G}_L = \{(x_i, y_i)\}_{i=1}^{N_L}$ be the labeled local dictionary with target vocabulary $\mathcal{Y}$, where x_i is a reference video, $y_i \in \mathcal{Y}$ is its gloss, and N_L is the

number of local references. Given a query video q, the task is to rank every $y \in \mathcal{Y}$. Because almost every target label is a singleton, CARE retains the training videos as reference instances instead of learning a separate classifier output for every target label.

CARE uses three retrieval branches that produce five score channels: one local Video-Swin channel, raw and aligned I3D channels, and raw and aligned SignRep channels. The external channels compare the query with a CISLR reference bank, whereas the local channel scores every target label. Since the external lexicon covers only part of $\mathcal{Y}$, these scores are fused through an explicit label-coverage mask. We precompute the representations and fit closed-form orthogonal alignment maps without updating the encoders. Fusion weights are selected using validation scores. Figure 1 summarizes the retrieval and fusion pipeline.

3.2 Representations and External Coverage

For Video-Swin (Liu et al., 2022) and WLASL-adapted I3D (Carreira and Zisserman, 2017; Li et al., 2020), each WR video is represented by a uniform temporal view and a motion-quantile view. The latter samples according to smoothed frame-difference energy, emphasizing frames in higher-motion regions. For SignRep (Wong et al., 2025), we process overlapping windows and pool their embeddings using predicted hand activity to downweight windows with low predicted activity. All resulting embeddings are ℓ_2-normalized and all encoders remain frozen. For a WR video z, let $h_b^v(z)$ denote its normalized embedding from encoder $b \in \{S, I, R\}$ under view v, where S, I, and R denote Video-Swin, I3D, and SignRep, respectively. For Video-Swin and I3D, $v \in \{u, m\}$ denotes the uniform or motion-quantile view; for SignRep, $v = \mathrm{act}$ denotes its single activity-pooled view. Accordingly, $\mathcal{V}_I = \{u, m\}$ and $\mathcal{V}_R = \{\mathrm{act}\}$. For a retained CISLR video p, let $h_b^E(p)$ denote its normalized external embedding for $b \in \{I, R\}$. We use $\mathrm{norm}(a) = a/\|a\|_2$.

We construct the external bank by normalizing WR and CISLR glosses through Unicode decomposition, lowercasing, and non-alphanumeric replacement, followed by unambiguous exact matching. Let $\mathcal{E}_y$ be the retained CISLR videos for target label y, and let $\mathcal{C} \subseteq \mathcal{Y}$ contain the labels with both I3D and SignRep external features. Every matched video remains an individual prototype; no class averaging is used during retrieval.

3.3 Local Prototype Retrieval

Video-Swin defines the complete local expert. For each label, CARE selects the closest local instance after taking the strongest agreement among the four uniform/motion view pairs:

$$D_{S,L}(q, y) = \min_{i:y_i=y} \Big[1 - \max_{v,w\in\{u,m\}} \langle h_S^v(q), h_S^w(x_i)\rangle \Big]. \quad (1)$$

This reduces to nearest-neighbor retrieval for singleton labels and remains valid for the few labels with multiple references. Here, L denotes the local bank and $\langle\cdot,\cdot\rangle$ is the inner product, which equals cosine similarity because the embeddings are normalized. The final system uses no additional local SignRep score.

3.4 Cross-Dataset Alignment and External Retrieval

WR and CISLR can occupy different regions of the same pretrained feature space because of differences in signers, framing, temporal boundaries, and recording conditions. For encoder $b \in \{I, R\}$, we compute normalized local and external centroids for every shared gloss $y \in \mathcal{C}$. The matrices $X_{b,v}$ and Y_b stack the paired local-view and external centroids, respectively, and $\mu_{b,v}^L$ and μ_b^E denote their row means. Here, $v \in \{u, m\}$ for I3D, while SignRep has one activity-pooled channel. Let $R_{b,v}^\star$ be the standard orthogonal Procrustes solution obtained by singular value decomposition after centering the two matrices (Schönemann, 1966). A local query is mapped into the external coordinate system as

$$\widetilde{h}_b^v(q) = \mathrm{norm}\big((h_b^v(q) - \mu_{b,v}^L)R_{b,v}^\star + \mu_b^E\big). \quad (2)$$

CARE retains both the original and aligned geometries. Let $h_{b,\mathrm{raw}}^v(q) = h_b^v(q)$ and $h_{b,\mathrm{align}}^v(q) = \widetilde{h}_b^v(q)$. External instance similarities are reduced to label distances by nearest-reference matching for $r \in \{\mathrm{raw}, \mathrm{align}\}$:

$$D_{b,E}^r(q, y) = 1 - \max_{p\in\mathcal{E}_y} \max_{v\in\mathcal{V}_b} \big\langle h_{b,r}^v(q), h_b^E(p)\big\rangle, \quad y \in \mathcal{C}. \quad (3)$$

Here, E denotes the external bank. Thus, class centroids determine only the alignment map, while retrieval retains individual external instances rather than class averages.

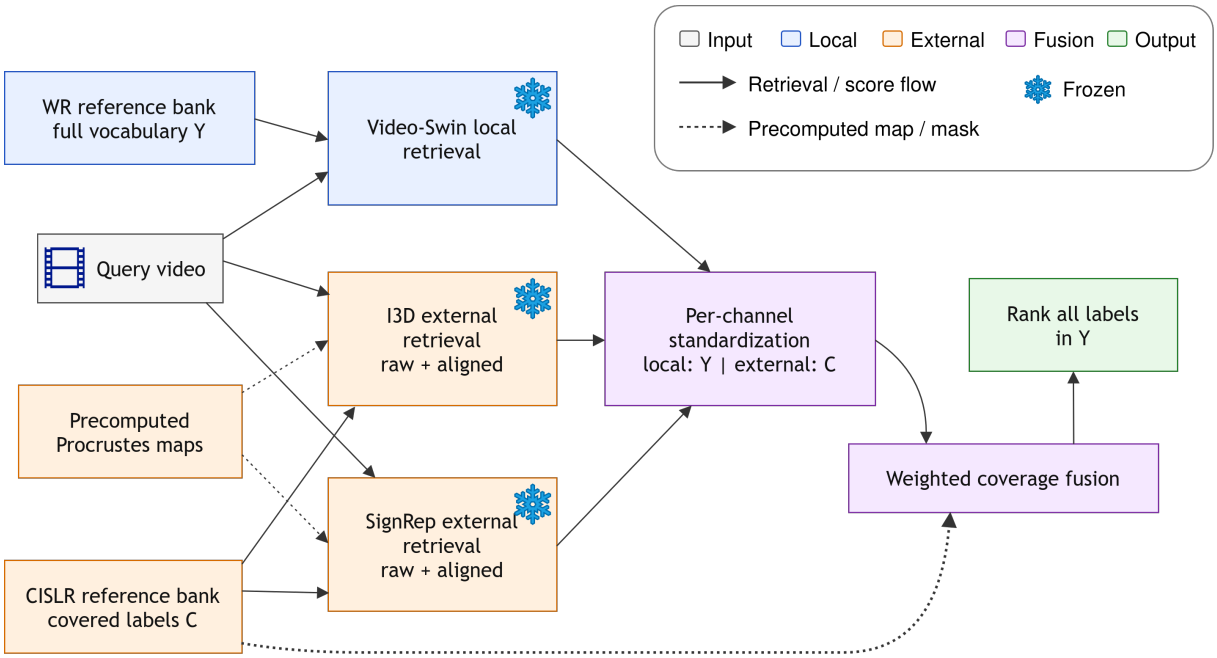


Figure 1: Overview of CARE.

3.5 Coverage-Aware Retrieval Ensemble

Because the retrieval channels can have different distance distributions, each is standardized independently for every query. For a distance function F with finite label support $\mathcal{A}$, $Z_{\mathcal{A}}[F](q, y)$ subtracts $\mathrm{median}_{y' \in \mathcal{A}} F(q, y')$ from $F(q, y)$ and divides by $\mathrm{std}_{y' \in \mathcal{A}} F(q, y') + \varepsilon$, where $\varepsilon > 0$ prevents division by zero. Local scores are standardized over $\mathcal{Y}$, whereas each external channel is standardized separately over $\mathcal{C}$. The complete fusion rule is

$$D(q, y) = Z_{\mathcal{Y}}[D_{S,L}](q, y) + \mathbb{I}[y \in \mathcal{C}] \sum_{b \in \{I,R\}} \sum_{r \in \{\mathrm{raw,align}\}} \lambda_b^r Z_{\mathcal{C}}[D_{b,E}^r](q, y). \quad (4)$$

Here, λ_b^r is the weight of external channel (b, r) and $\mathbb{I}[\cdot]$ is the indicator function. The indicator is the coverage-aware component: external evidence is added only to labels with valid CISLR support, while uncovered labels retain their complete local score. The term ensemble denotes the late fusion of these normalized score channels. At inference, labels are ranked in ascending order of $D(q, y)$; no backbone update or target-vocabulary classifier is required.

4 Experiments

4.1 Datasets

WSLP 2026 ISLR (WR). The dataset was released for the Isolated Sign Language Recognition track of the WSLP 2026 Shared Task on Indian Sign Language Processing. It contains 5,033 videos: 4,398 labeled training examples, 109 validation queries, and 526 official test queries. Its training vocabulary contains 4,361 glosses, of which 4,336 (99.43%) are singletons. Validation and test labels are withheld, and predictions are evaluated by the official WSLP 2026 task server.

CISLR. We use the public CISLR corpus (Joshi et al., 2022) only as labeled external data. The corpus contains 7,050 videos and 4,765 glosses. The lexical alignment defined in the Methodology retains 3,739 videos covering 2,016 target labels (46.23%). This bank comprises 2,017 videos from the CISLR prototype partition and 1,722 from its released test partition; we do not evaluate performance on the original CISLR task.

4.2 Evaluation Metrics

Following the official WSLP 2026 evaluation metrics, performance is measured using Top-1, Top-5, and Top-10 accuracy. Top-K accuracy is the percentage of videos for which the ground-truth gloss appears among the first K predictions. Top-1 therefore measures exact recognition, whereas Top-5

and Top-10 measure the quality of the short ranked candidate list returned by the system.

4.3 Implementation Details

Video preprocessing and local retrieval. All RGB frames are resized to 224×224 pixels. For Video-Swin, each uniform and motion-quantile view contains 32 frames. We use a frozen Video-Swin-S model initialized with its default pretrained weights. We remove the classification head and apply the normalization associated with those weights.

I3D features. The I3D branch uses the public RGB checkpoint fine-tuned on WLASL-2000. Each WR view contains 96 frames. We extract the final spatio-temporal feature sequence, normalize each temporal vector, average over time, and normalize the result again. External CISLR I3D sequences are pooled using the same normalize–average–normalize rule.

SignRep features. We use the released SignRep checkpoint with its backbone frozen. Videos are divided into 16-frame windows with a stride of 8 frames. Window embeddings are pooled using predicted hand activity to downweight windows with low predicted activity. CARE uses the resulting activity-weighted representation.

Alignment and score fusion. We fit one Procrustes map for each I3D view and one for SignRep using the complete shared-label set. We set $\varepsilon = 10^{-6}$ and use $\lambda_I^{\text{raw}} = 3.25$, $\lambda_I^{\text{align}} = 3.50$, $\lambda_R^{\text{raw}} = 1.00$, and $\lambda_R^{\text{align}} = 6.00$. We assign the local Video-Swin score a weight of 1.00.

5 Results

5.1 Validation and Test Performance

Table 1 reports the performance of the final CARE configuration. Official test accuracy is lower than validation accuracy by 8.72%, 9.04%, and 7.90% at Top-1, Top-5, and Top-10, respectively. Although all backbones remain frozen, repeated validation-based selection of the fusion weights and system configuration may have introduced selection bias. Sampling variability and differences between the split distributions may also contribute to the gap; the aggregate scores alone do not establish their relative contributions. On the test split, Top-10 accuracy exceeds Top-1 accuracy by 2.66%, corresponding to 14 additional queries with the correct label at ranks 2–10.

5.2 Archived External Encoder Comparison

Table 2 compares the two external configurations; the second row corresponds to the final CARE system in Table 1. The SignRep-based configuration scores 2.75% higher at Top-1 and 1.84% higher at both Top-5 and Top-10 than the Video-Swin-based configuration. Because the fusion weights of the two configurations were selected separately on the same validation split, this is a descriptive comparison rather than a controlled ablation.

6 Conclusion

We presented CARE, which combines a complete local dictionary with raw and aligned external retrieval under a coverage mask. Without target-specific backbone fine-tuning, it achieves 65.59% Top-1 accuracy on the official 526-query WSLP 2026 test split. The validation–test gap warrants caution when interpreting performance on the small validation split used for model selection. The reported comparison does not isolate the contributions of individual components; a controlled ablation remains necessary to assess them.

References

Matyáš Boháček and Marek Hrúz. 2023. Learning from what is already out there: Few-shot sign language recognition with online dictionaries. In *2023 17th IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, pages 1–6.

Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pages 4724–4733. IEEE.

Mathieu De Coster and Joni Dambre. 2023. Querying a sign language dictionary with videos using dense vector search. In *2023 IEEE International Conference on Acoustics, Speech and Signal Processing Workshops (ICASSPW)*, pages 1–5.

Aashaka Desai, Lauren Berger, Fyodor Minakov, Nessa Milano, Chinmay Singh, Kriston Pumphrey, Richard E. Ladner, Hal Daumé III, Alex X. Lu, Naomi Caselli, and Danielle Bragg. 2023. ASL Citizen: A community-sourced dataset for advancing isolated sign language recognition. In *Advances in Neural Information Processing Systems*, volume 36. Curran Associates, Inc.

Hezhen Hu, Weichao Zhao, Wengang Zhou, Yuechen Wang, and Houqiang Li. 2021. Signbert: Pre-training of hand-model-aware representation for sign language recognition. In *2021 IEEE/CVF Interna-*

Table 1: CARE results on the WSLP 2026 validation and official test splits. Values in parentheses give the number of queries for which the correct gloss appears within the first K positions.

Split	Queries	Top-1 (%)	Top-5 (%)	Top-10 (%)
Validation	109	74.31 (81)	76.15 (83)	76.15 (83)
Official test	526	65.59 (345)	67.11 (353)	68.25 (359)

Table 2: Validation performance of two archived external-encoder configurations. Values in parentheses give the number of queries whose correct gloss appears within the first K positions.

External branches	Top-1 (%)	Top-5 (%)	Top-10 (%)
I3D + Video-Swin	71.56 (78)	74.31 (81)	74.31 (81)
I3D + SignRep	74.31 (81)	76.15 (83)	76.15 (83)

tional Conference on Computer Vision (ICCV), pages 11067–11076. IEEE.

Abhinav Joshi, Ashwani Bhat, Pradeep S, Priya Gole, Shashwat Gupta, Shreyansh Agarwal, and Ashutosh Modi. 2022. CISLR: Corpus for Indian Sign Language recognition. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10357–10366, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. 2020. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1459–1469.

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. 2022. Video swin transformer. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 3192–3201. IEEE.

Peter H. Schönemann. 1966. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10.

Prem Selvaraj, Gokul Nc, Pratyush Kumar, and Mitesh Khapra. 2022. OpenHands: Making sign language recognition accessible with pose-based pretrained models across languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2114–2133, Dublin, Ireland. Association for Computational Linguistics.

Jake Snell, Kevin Swersky, and Richard S. Zemel. 2017. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 30, pages 4077–4087.

Baochen Sun, Jiashi Feng, and Kate Saenko. 2016. Return of frustratingly easy domain adaptation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1):2058–2065.

Hamid Reza Vaezi Joze and Oscar Koller. 2019. MS-ASL: A large-scale data set and benchmark for understanding american sign language. In *Proceedings of the British Machine Vision Conference*.

Toon Vandendriessche, Mathieu De Coster, Annelies Lejon, and Joni Dambre. 2025. Representing signs as signs: One-shot ISLR to facilitate functional sign language technologies. *arXiv preprint arXiv:2502.20171*.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. 2016. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, volume 29, pages 3630–3638.

Chang Wang and Sridhar Mahadevan. 2008. Manifold alignment using procrustes analysis. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1120–1127.

Ryan Wong, Necati Cihan Camgoz, and Richard Bowden. 2025. SignRep: Enhancing self-supervised sign representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22804–22814.

Kaichao You, Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. 2019. Universal domain adaptation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2715–2724. IEEE.

Bukva: Russian Sign Language Alphabet

Petr Surovtsev[1] pysurovtsev@daimld.tech
Alexander Nagaev[1] aonagaev@daimld.tech
Karina Kvanchiani[1] karinakvanciani@gmail.com
Elizaveta Petrova[1] kleinsbotle@gmail.com
Alexander Kapitanov[1] aakapitanov@daimld.tech

[1]Department of Artificial Intelligence and Machine Learning Development, DAIMLD

Abstract

This paper investigates the recognition of the Russian fingerspelling alphabet, also known as the Russian Sign Language (RSL) dactyl. Dactyl is a component of sign languages where distinct hand movements represent individual letters of a written language. This method is used to spell words without specific signs, such as proper nouns or technical terms. An alphabet learning simulator is one meaningful application of isolated dactyl recognition. There is a notable issue of data shortage in isolated dactyl recognition: existing Russian dactyl datasets lack subject heterogeneity, contain insufficient samples, or cover only static signs. We provide Bukva, the first open-source video dataset for RSL dactyl recognition that is free of all these shortcomings. It contains 3,757 videos with more than 100 samples for each RSL alphabet sign, including dynamic ones. We utilized crowdsourcing platforms to increase the subjects' heterogeneity, resulting in the participation of 155 deaf and hard-of-hearing experts in the dataset creation. We use a TSM (Temporal Shift Module) block to handle static and dynamic signs effectively, achieving 83.6% top-1 accuracy with real-time inference on CPU only. The dataset, demo code, and pre-trained models will be made publicly available[1].

1 Introduction

According to the World Health Organization, over 430 million people worldwide have disabling hearing loss[2], and many deaf and hard-of-hearing people use sign language. According to the statistical report for 2019, there are more than 150 thousand deaf sign language speakers in Russia[3]. Such languages are unique in terms of rules and grammar in different countries. A fundamental component of sign language is the SL alphabet called dactyl. It visually represents the words of the spoken language letter by letter, a process known as fingerspelling or dactylology. Fingerspelling spells words without a distinct sign, such as proper names and specialized terms, e.g., technical and scientific words and names of people or animals. Also, any word can be fingerspelled letter by letter.

Fingerspelling recognition is an urgent task with vast potential for practical applications. Such a system can be implemented at train stations and airports to indicate toponyms and in hospitals when describing diagnoses and medications. Schools, universities, and other institutions also require a dactylology recognition system to simplify communication using special terms. Another helpful application case is the creation of an interactive SL trainer. The user is shown an example of performing a sign, after which the recognition system checks whether the user repeated it correctly. Since there is a lack of sign language teachers and free courses with feedback, such trainers could improve accessibility to learning Russian Sign Language (RSL). Throughout this work, we consulted with members of the "All-Russian Society of the Deaf" (VOG), who confirmed that a dactyl simulator could simplify learning the alphabet for further usage in fingerspelling. A similar application exists for American Sign Language — the browser-based fingerspelling.xyz[4]. While fingerspelling applications involve continuous dactyl recognition, the SL trainer represents an isolated task since using a fingerspelling model would be redundant. This paper focuses on recognizing isolated letters to build the dactyl recognition system for a sign language alphabet trainer. It can also serve as a baseline for continuous fingerspelling recognition.

Moreover, fingerspelling is a key bottleneck for sign language translation (SLT): proper nouns,

[1]https://github.com/ai-forever/bukva
[2]https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss
[3]https://voginfo.ru/about/
[4]https://fingerspelling.xyz

Proceedings of the Second Workshop on Sign Language Processing, pages 147–161
October 29, 2026 ©2026 Association for Computational Linguistics

named entities, and out-of-vocabulary technical terms are conveyed almost entirely through the alphabet, and these are precisely the tokens that most often break gloss-free SLT in low-resource settings. Robust isolated-letter recognition is therefore a building block for downstream continuous SLT, not only an educational tool.

Some unique challenges are presented in the dactyl recognition task. It involves smaller, faster movements that are more difficult to recognize than regular sign language. Since dactyl movements are short and quick, motion blur can have a significant negative impact, requiring the network to be more robust to alterations in data quality. The presence of static and dynamic letters in numerous dactyl alphabets makes it impossible to create a dataset containing only images. Moreover, some dactyl letters in RSL can appear similarly, making recognition problematic. For instance, "л" ("l") – "п" ("p") and "м" ("m") – "т" ("t") have comparable configurations and are distinguished only by the spacing between fingers. Similarly, "ё" ("yo") differs from "е" ("e") solely in the hand movement.

Given the use cases discussed above, it is clear that the dactyl recognition system must be robust to variations in the background, lighting, and subjects. It must also operate in real-time and handle videos of varying quality and resolution. Striving for high quality is also crucial, as even a small error can have significant consequences. These requirements place certain constraints on the model: it must be efficient enough for real-time processing and of acceptable size to avoid overfitting caused by the relatively small number of classes in the dactyl alphabet. Furthermore, the task's specifics restrict the training data: the dataset must be heterogeneous across subjects and scenes, balanced across classes, and contain sufficient samples. Existing Russian dactyl datasets are unsuitable because they lack diverse subjects and scenes and sufficient samples. Besides, some cannot be used to build a full-fledged SL alphabet recognition system as they only cover static signs, omitting dynamic ones.

This work presents Bukva, the first full-fledged open-source video dataset for classifying the RSL alphabet. By full-fledged, we mean a dataset that simultaneously (1) covers all 33 RSL alphabet letters, including dynamic ones, (2) provides a sufficient number of samples per sign (more than 100), and (3) features heterogeneous signers, scenes, and recording conditions; none of the existing RSL alphabet datasets, including the open-source Slovo (Kapitanov et al., 2023), satisfies all these criteria. RSL remains an underrepresented sign language in terms of available resources. Since RSL significantly differs from the Russian language, we regularly consult with RSL teachers and native speakers to create the data, considering the language's rules and nuances. The resulting dataset contains 3,757 high-quality videos divided into 33 classes. The videos were captured using various devices and featured 155 RSL speakers as signers, showcasing diverse scenes and lighting conditions. We integrated the RSL exam into the data creation pipeline to limit video recording access to individuals without RSL knowledge. Besides, all videos were filtered and validated to ensure the quality of the data. Bukva also includes an extra "no sign" class to assist the model in determining the sign frames from the entire video.

In experiments, we employ the TSM block (Lin et al., 2019) to balance different frame variability according to the temporal component for static and dynamic signs. It enables the 2D convolutions to process video data without increasing the number of parameters. The experiments demonstrate that a model trained using the proposed dataset achieves high metrics, with a top-1 accuracy of 83.6% and real-time CPU performance. Furthermore, we also provide a demo application illustrating the operation of a simulator for teaching dactyl in Russian sign language. The Bukva dataset, demo code, and pre-trained models will be made publicly available.

Our contributions can be summarized as follows: (1) Bukva involves the largest number and diversity of signers among RSL alphabet datasets (155); (2) all samples are HD or FullHD videos; (3) the dataset covers all 33 RSL alphabet letters, including dynamic ones; (4) the dataset, demo code, and pre-trained models are open-sourced; (5) we describe a reproducible crowd-based collection and annotation methodology — proficiency-gated recruitment, multi-annotator validation, and sign-interval annotation — transferable to other low-resource sign languages

2 Related Work

This section overviews widespread sign language alphabet datasets. We prioritize isolated alphabetic RSL datasets and cover fingerspelling and isolated datasets in other languages only briefly to focus the reader on the primary topic. Note, however, that datasets of other sign languages can be helpful,

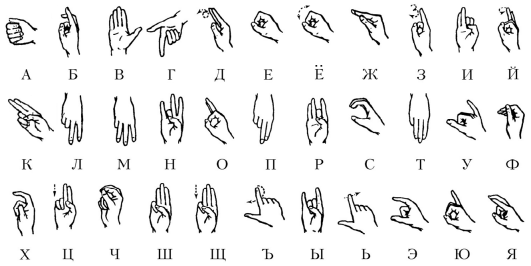


Figure 1: Each of the RSL alphabet's 33 signs corresponds to a letter of the Russian language alphabet. Arrows indicate the direction of movement for 8 dynamic signs.

for example, for pre-training or developing multilingual sign language models. Since one of this paper's main parts is the data creation process, we also analyze existing methods for collecting and annotating dactyl data, regardless of language.

2.1 Russian Sign Language Alphabet Datasets

To our knowledge, only three datasets contain RSL alphabet isolated signs. Two are dedicated to dactyls, while one contains alphabet signs on par with other word signs. Makarov et al. (2019) proposed a dataset of the RSL alphabet divided into 26 static signs, while samples for the remaining 7 dynamic RSL signs were decided not to be collected[5]. This choice allowed the authors to create a dataset consisting of only images. 20 people in 20 different scenes recorded 520 samples subsequently cropped by authors. The experiments with this dataset were redundant since it is limited to static images and does not contain dynamic signs. Grif and Kondratenko (2021) collected 14,978 samples, images for 25 static signs, and videos for 8 dynamic ones. 4 people participated in the recording in studio conditions. The comparison analysis with this dataset was impossible since it is private. Kapitanov et al. (2023) presented the dataset Slovo – the most significant open dataset of isolated RSL signs. Among 20,000 videos, only 660 were provided for 33 dactyl signs, recorded by 30 people. The Slovo was collected on crowdsourcing platforms with significant heterogeneity in recording conditions.

Considered datasets are unsuitable for building a dactyl recognition system limited by conditions in Section 1. Samples from Makarov et al. (2019) were collected only for static signs. Moreover, they are artificially deprived of context using cropping. Also, on par with data from Grif and Kondratenko (2021), the images are homogeneous in subjects, negatively affecting model overfitting. The Slovo dataset's number of samples is insufficient to train a high-quality model. However, the Slovo is an appropriate basis for constructing a more powerful dataset. In contrast to these datasets, Bukva provides (1) the largest number and diversity of signers (155 vs. 4–30), (2) HD/FullHD video samples, and (3) coverage of all 33 RSL alphabet letters, including dynamic ones, in an open-source dataset.

2.2 Other Sign Language Alphabet Datasets

There are multiple dactyl datasets consisting only of static signs. The American SL (ASL) dataset (Kang et al., 2015) includes 24,000 depth maps divided into solely 24 static signs, despite dynamic signs in ASL. Analogously, static signs entirely fill the Japanese SL dataset (Hosoe et al., 2017) and the Bengali alphabet dataset (Podder et al., 2020). Another group of dactyl datasets represents dynamic signs with images. MU HandImages ASL (Barczak et al., 2011), the Polish (Szyszka, 2021), and Kazakh (Kenshimov et al., 2021) dactylic datasets consist of images for static and dynamic signs. In these datasets, dynamic signs are represented by an image of the hand's final position. The most appropriate choice of modality for dactyl data is video. Such datasets as the RWTH-Gesture Database (Dreuw et al., 2006), the KArSL Arabic SL dataset (Sidig et al., 2021), and the KSU-SSL dataset for the Saudi language (Alsulaiman et al., 2023) contain video samples for all signs. This subsection also considers fingerspelling datasets to show the difference between continuous and isolated dactyl tasks. The dataset from Kim et al. (2016) consists of 3,684 sign sequences of 21,453 letters. The ChicagoFSWild dataset (Shi et al., 2018) and its extension (Shi et al., 2019) contain 62,536 sign sequences and 302,806 letters. The largest continuous fingerspelling dataset is provided by Google – American Sign Language Fingerspelling Recognition competition (Chow et al., 2023); note that it differs in modality, as its samples are hand keypoints rather than videos.

2.3 Data Collection Pipeline

Considered datasets can be categorized based on (1) samples' recording method, (2) sign bucket choice, (3) expert choice, (4) additional modalities, and (5)

[5]Based on expert consultation, 8 letters are dynamically displayed, but due to uncertainty in the RSL alphabet, the authors of Makarov et al. (2019) consider only 7 letters are shown with movement.

Table 1: The main parameters of widespread SL alphabets in different languages. The datasets are divided into two groups, containing predominantly images or videos. Relevant Russian SL datasets are highlighted in blue. In the Resolution column, "crop" denotes images cropped to the hand region, and "signals" denotes data glove sensor recordings rather than RGB samples. The WLASL row refers only to the alphabet-letter subset of this word-level dataset.

Dataset	Letters	Samples	Signers	Resolution	Language
Images					
MU HandImages ASL, 2011 (Barczak et al., 2011)	26	2,425	5	crop	American
ASL Fingerspelling, 2015 (Kang et al., 2015)	24	24,000	5	320 × 240	American
RSL dactyl, 2019 (Makarov et al., 2019)	26	520	20	crop	Russian
BdSL alphabet, 2021 (Podder et al., 2020)	37	35,149	350	crop	Bangla
PSL Alphabet 100, 2021 (Szyszka, 2021)	36	3,707	–	crop	Polish
Kazakh Sign Language, 2021 (Kenshimov et al., 2021)	42	2,100	–	crop	Kazakh
Videos					
RWTH-Gesture Database, 2006 (Dreuw et al., 2006)	30	1,200	20	varying	German
WLASL, 2020 (Li et al., 2020)	21	150	12	varying	American
KArSL, 2021 (Sidig et al., 2021)	39	5,850	3	512 × 424 / FullHD	Arabic
RSL alphabet (NSTU), 2021 (Grif and Kondratenko, 2021)	33	14,978	4	varying	Russian
PSL Alphabet, 2023 (Piskozub, 2023)	36	43,200	16	signals	Polish
AzSL dactyl alphabet, 2023 (Hasanov et al., 2023)	32	13,440	221	varying	Azerbaijani
KSU-SSL, 2023 (Alsulaiman et al., 2023)	37	18,315	33	varying	Saudi
Slovo, 2023 (Kapitanov et al., 2023)	33	660	30	HD / FullHD	Russian
NSL23, 2024 (Sunuwar et al., 2023)	49	1,205	23	varying	Nepali
Bukva, 2026 (ours)	33	3,757	155	HD / FullHD	Russian

annotation method.

Recording method One commonly used method for gathering dactyl samples is through studio recording. The dataset from Barczak et al. (2011) was recorded in a studio with a green background and with specially designed moving light sources to create varied lighting effects. The authors of the papers (Kang et al., 2015; Kim et al., 2016; Dreuw et al., 2006) also selected a studio recording. Some papers utilized the attempts to diversify the data recorded in studios. So, the KSU-SSL videos (Alsulaiman et al., 2023) were recorded by simultaneously employing three cameras, while in Hosoe et al. (2017), the recorders rotated and changed poses during the sign display. The background color was changed while recording the NSL23 dataset (Sunuwar et al., 2023). The combination of studio recording and internet downloads is also widespread in the sign video collection. Grif and Kondratenko (2021) constructed a dataset of 1,566 samples from YouTube and 13,412 samples recorded in the studio. Shi et al. (2018) proposed a dataset consisting entirely of YouTube videos, while its extended version (Shi et al., 2019) was supplemented with videos recorded in the studio. The authors of several datasets received samples through crowdsourcing. Thus, in Makarov et al. (2019), part of the data was recorded by 15 people in real conditions and transferred to the authors online. Videos of dactyl signs for the Slovo dataset (Kapitanov et al., 2023) were obtained from 30 people through a crowdsourcing platform. The authors validated the data for errors in showing signs due to the perception of crowdworkers as low-skilled.

Sign bucket choice Depending on whether the selected signs were dynamic or not, the authors decided on the modality of the final dataset: (1) video for dynamic and images for static, (2) video for all signs (if there are dynamic ones), (3) images for all signs (if there are no dynamic ones). However, there are two exceptions: (1) the authors from Barczak et al. (2011) collected images for all dactyl signs of the ASL, including a number for dynamic ones; (2) the authors from Makarov et al. (2019) and Kang et al. (2015) collected samples only for static signs, ignoring dynamic ones to simplify the data collection process.

Expert choice Since recording sign videos requires specialized knowledge, selecting signers is integral to creating SL data. The authors of the papers (Kim et al., 2016; Sunuwar et al., 2023) involved native SL speakers and beginners just learning it. Videos from the KSU-SSL dataset (Alsulaiman et al., 2023) were obtained from individuals specially trained in sign language, followed by expert moderation. The Slovo dataset authors (Kapitanov et al., 2023) restricted video recording task access to signers who had not passed the RSL

exam. The exam was structured to only be passable by native speakers and students of the RSL.

Additional modalities Numerous datasets (Sunuwar et al., 2023; Makarov et al., 2019; Kenshimov et al., 2021; Hosoe et al., 2017; Szyszka, 2021; Podder et al., 2020; Dreuw et al., 2006; Kapitanov et al., 2023) represent images or videos showing hands performing a sign. However, some authors used specialized devices such as Microsoft Kinect V2 (Sidig et al., 2021) and Creative Senz3D (Kang et al., 2015) to collect depth maps for each sign. Also, Microsoft Kinect V2 allowed the authors (Sidig et al., 2021) to obtain human skeleton annotations. The authors from Hosoe et al. (2017) made 3D sign models to receive 8,000 synthetic images of signs from different angles.

Annotation method Considered datasets practically avoid additional annotation since samples contain a single sign. However, the Slovo (Kapitanov et al., 2023) consists of extra videos that include normal human movement. Such "sign free" sequences can further assist the model in detecting signs in a sequence of frames.

3 Dataset Creation Pipeline

We collected videos for all 33 RSL letters via the ABC Elementary and Yandex Toloka crowdsourcing platforms. Following Kapitanov et al. (2023), the pipeline comprises three stages: (1) video collection, (2) video filtering and validation, and (3) sign time interval annotation. All crowdworkers passed a mandatory exam to verify RSL proficiency: scoring 80% or higher granted access to video validation, and 90% or higher to video recording.

Of approximately 8,400 collected videos, we filtered out low-quality samples and duplicates (8.6% in total), normalized the rest to 30 FPS, and approved about 6,000 (71.4%) in validation by at least three workers per video. Three annotators then marked the sign-showing interval in each video. Finally, we combined the approved samples with 660 videos from the Slovo dataset, ensuring at least 101 unique samples per class, and discarded videos where the hand left the frame. See Appendix A for details, including compensation, exam examples, and the validation and annotation interfaces.

4 Dataset Characteristics

Dataset Versions We build two versions of the Bukva: an untrimmed version and a trimmed version with an extra 34th class "no sign" with 23 GB and 3.2 GB sizes, respectively. 105 additional samples were randomly obtained by annotations from Section A.4 to expand the second set. There is a further description of the initial untrimmed version for clarity. The explicit "no sign" class, together with the sign time-interval annotations, additionally supports sign-presence prediction and the temporal localization of fingerspelling within continuous signing. Detailed statistics for each class label are provided in Appendix B.

Dataset Content The Bukva dataset is a collection of 3,757 videos representing the RSL alphabet. It is divided into 33 signs, each corresponding to a letter of the Russian alphabet. On average, 113 videos per class with an average length of 49 frames were recorded by 155 signers with RSL knowledge. Some participants communicate with sign language daily due to hearing limitations, while others are RSL learners or teachers. The dataset presents videos showing signers aged from $\sim$9 to $\sim$81. The information about including videos with children in the dataset can be found in Section 6. Most of the videos (80%) were recorded by women, presumably because most RSL teachers and translators are women. The dataset was primarily gathered indoors and included various scenes and lighting conditions. Figure 2 shows the distance to the camera and the video's brightness distributions. Because we release per-signer identifiers together with demographic (age, gender) and acquisition metadata (device, resolution, distance, brightness), Bukva also enables fairness and robustness analyses across signer subgroups and recording conditions, which we encourage the community to explore.

Dataset Size and Quality About 70% of the Bukva videos are in FullHD format. Signers recorded the videos using personal devices such as smartphones and laptops (see Figure 2). Since different devices record videos with varying FPS, we standardized all videos to a consistent FPS. 81% of the videos are in portrait orientation, 16% are in landscape orientation, and 3% are in square format. The total duration of the dataset is approximately 4 hours.

Dataset Splitting The Bukva was divided into train and test sets, comprising 3,097 (82%) and

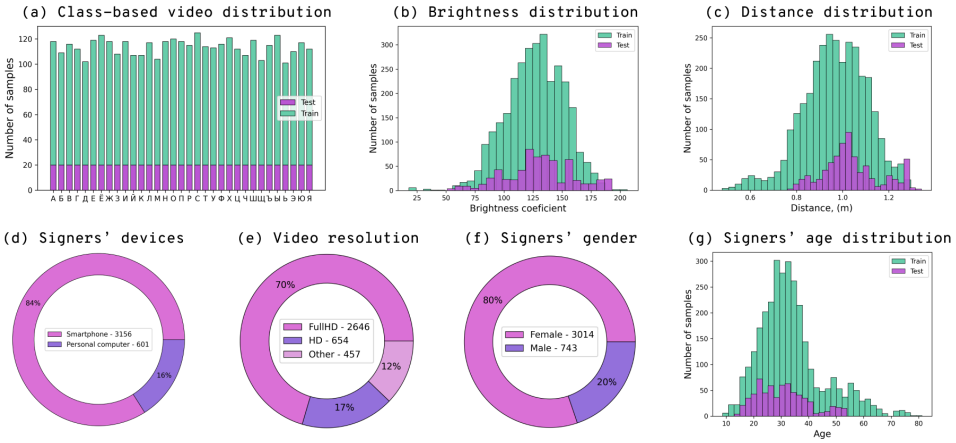


Figure 2: Video resolution, brightness, signers, classes, and splitting analysis. a) class-based video distribution: number of videos for each RSL alphabet letter in the train and test sets; b) brightness distribution: the central image of each video sign part is extracted, converted to grayscale, and then the average pixel brightness is calculated; c) distance distribution: the distance is approximately estimated in meters by computing the length between the signers' left and right shoulders using MediaPipe (Zhang et al., 2020); d) signers' devices; e) video resolution; f) signers' gender; g) signers' age distribution: the age is determined by the MiVOLO model (Kuprashevich and Tolstykh, 2023).

660 (18%) videos, respectively. We intentionally designed the sets to ensure no overlap between the signer groups. Twenty signers who recorded videos for all 33 signs were selected to construct the test set. The train part contains 81 to 105 videos for each sign.

5 Experiments

Since we are dealing with static and dynamic signs recorded as videos, it is necessary to find a solution to balance image and video classification. Model capacity for image classification is insufficient to cope with dynamics, while 3D architectures are redundant for statics. The TSM block (Lin et al., 2019) effectively handles this task by modifying the 2D convolutions operation without changing the model architecture and increasing the parameters. Such features enable us to use a pre-trained model with weights from widespread image classification models for the dactyl recognition task. We apply the TSM block to train three architectures: (1) the lightest and fastest MobileNetV2 (Sandler et al., 2019), (2) MobileOne (Vasu et al., 2023) from S0 to S2 for a balance between speed and number of parameters, and (3) two ResNet versions (He et al., 2015) with 18 and 50 blocks as the most widespread image encoders.

5.1 Experiments Setup

An essential aspect of training video encoders is a specific strategy for selecting frames from videos. We sample 8 frames uniformly with an individual step calculated by the formula $step = round(\frac{V_{len}}{N})$, where V_{len} is the video length, N is the frame number for sampling, the result is rounded down. This strategy allows us to effectively train models with weak receptive fields since each sampled frame contains a certain proportion of dynamic signs.

We utilize a unified setup for all architectures to facilitate transparent comparisons. An SGD optimizer with an initial learning rate of 0.02, momentum of 0.9, and weight decay of 0.00002 is fixed in the setup. The learning rate is modified by a LinearLR scheduler up to the 20th epoch and then by CosineAnnealingLR (Loshchilov and Hutter, 2017) until the end of training. The models are trained on two Tesla V100 with 32GB VRAM, using a batch size of 16 for each card. It is sufficient for all models to converge in 80 epochs.

5.2 Dataset Capacity Ablation

Reducing the training set size from 3,182 to 2,000 samples undeniably resulted in an apparent decline in top-1 accuracy, from 0.8368 to 0.7956. A further reduction to 1,000 samples resulted in a significant drop to 0.5676. The number of signers was kept consistent across all three experiments. Similarly, reducing the number of unique signers from 70 to 50 with consistent 1,500 samples decreases the top-1 accuracy from 0.7588 to 0.7415. These results prove an unquestionable decline in top-1 when ei-

ther the number of samples per class or the diversity of users is reduced. Also, this emphasizes the significance of having a balanced dataset to guarantee strong model performance.

5.3 Data Preprocessing

The experiments are based on the trimmed Bukva version with an extra "no sign" class, allowing the model to learn to separate signs from other hand movements. Eighty-five samples expand the training set for an additional class, while the test set is expanded by 20, resulting in 3,182 and 680 videos, respectively.

Training Set. The vertical and horizontal videos are required to be squared for model input. Primarily, the source video frames are resized to the smaller side of 300 to preserve the aspect ratio. Further, we randomly employ either multi-scale cropping or padding instead of straightforward square resizing to prevent unrealistic spatial stretching of the video. Since dactyl is invariant to the hand side, we reflected the frames vertically by flip augmentation with a probability of 0.5. Multi-scale cropping diversifies the training set regarding hand size and position in the frame, while the flip balances videos on hand orientation. **Test Set.** Each frame is resized to 224 on the smaller side and padded with 114 to square. We processed the test set without cropping to maintain the dataset's original distribution.

5.4 Results

Table 2 presents the results of evaluating models using the top-1 accuracy metric. We also provide information about the model sizes, the number of parameters, and the inference time to distinguish the optimal trade-off between quality, model size, and prediction speed. Note that the test set contains only 20 videos per class, so the limited amount of test data results in high error costs and could lead to overestimating model performance. Since ResNet architecture is high-capacity, the small amount of training data and 76% static signs in the RSL alphabet contributed to model overfitting. Meanwhile, MobileNet performs better due to its relatively compact size. There are multiple false positives in the predictions for similar static signs (see Figure 3). For example, the letters "н" ("n") and "р" ("r") appear almost identically (see Figure 1), so the model is frequently mistaken while predicting each of them. The confusion matrix shows that the model struggles to differentiate visually similar signs: errors occur between the dynamic sign "щ" ("shch") and the static sign "ш" ("sh"), which have similar finger and hand positions, as well as between signs with similar shapes, such as "п" ("p"), "т" ("t"), "м" ("m"), and "л" ("l"). At the same time, the model reliably distinguishes visually contrasting signs. These results suggest that improvements should target sensitivity to subtle differences in hand shapes and movements.

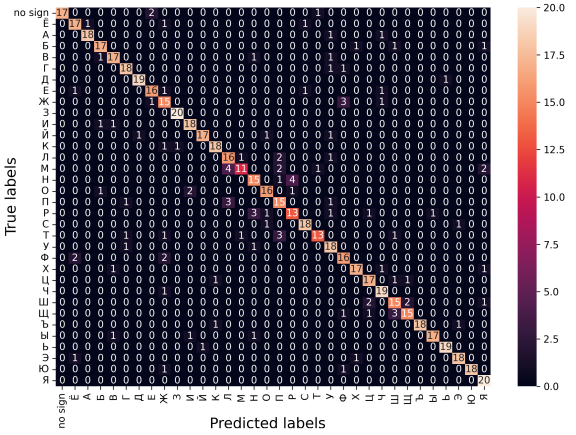


Figure 3: The confusion matrix for test samples using the MobileNetV2 model. Off-diagonal cells indicate misclassifications. The model confuses the classes "м" ("m") and "л" ("l"), as their signs are nearly identical.

5.5 Application

Figure 4 shows the demo stand that provides real-time sign language recognition to learn RSL. The user can select any sign from the dataset and learn it offline after watching the corresponding video template. The trained classifier is integrated into the trainer as follows: (1) the user is presented with the correct version of a sign and prompted to replicate it; (2) the system recognizes the sign produced by the user; and (3) it provides a confidence score, which reflects the degree of accuracy in the user's execution of the sign. Thus, a learner who produces a sign slightly incorrectly receives a low confidence score as feedback rather than a simple acceptance of the closest class. A similar approach is used in the browser-based fingerspelling.xyz application for the ASL alphabet. An automatic system for learning any language cannot fully replace a teacher; however, due to the lack of available resources for studying RSL, a dactyl simulator can be a good starting point for building a comprehensive learning system.

Table 2: The models' training results on the Bukva. We evaluate models with top-1 accuracy as the standard metric in the action recognition task. All models are tested on a MacBook M1 Pro using a single core. We measured the model's inference 100 times with further averaging.

Model	Model Size (MB)	Parameters (M)	Inference Time (ms)	top-1 acc.
ResNet-18	44.8	11.2	191	72.7
ResNet-50	94.6	23.6	333	79.2
MobileNetV2	10.3	2.3	145	**83.6**
MobileOne S0	4.4	1.1	94	71.3
MobileOne S1	14.1	3.5	165	75.7
MobileOne S2	23.4	5.8	232	77.2



Figure 4: Russian sign language demo stand for education.

6 Conclusion

This paper presented Bukva, the first full-fledged open-source video dataset for isolated Russian Sign Language dactyl recognition. The dataset contains 3,757 HD and FullHD videos covering all 33 RSL alphabet letters, including dynamic ones, recorded by 155 signers – the largest number and diversity of signers among existing RSL alphabet datasets. The crowdsourcing-based creation pipeline with a mandatory RSL exam, multi-stage filtering, validation, and time interval annotation ensured the quality of the data. Baseline experiments with the TSM block demonstrated that a lightweight MobileNetV2 model achieves 83.6% top-1 accuracy with real-time CPU inference, while the ablation study confirmed the importance of both the number of samples and the diversity of signers. We also presented a demo application illustrating how the trained model can be integrated into an RSL alphabet learning simulator. More broadly, Bukva offers a reusable building block for continuous RSL recognition and translation, and a template for community-driven data collection in other underrepresented sign languages. The dataset, demo code, and pre-trained models will be made publicly available upon acceptance.

Limitations

Although the Bukva contains enough samples for training lightweight MobileNet, more is required to train ResNet architecture without overfitting. Also, the presence of static signs in the majority may cause such degradation of ResNet's metrics. However, since dactyl recognition applications require only real-time inference, the described problem is irrelevant.

Another limitation concerns the applicability of the proposed dataset to continuous fingerspelling recognition. Continuous fingerspelling includes a significant amount of coarticulation: the appearance of each letter depends on the preceding and following letters, and signs are displayed rapidly from various angles without transitions between them. To fully address the fingerspelling problem, a model must be trained to recognize rapid, subtle movements and the influence of neighboring letters on each letter's appearance. Consequently, while a model trained on the Bukva can serve as a baseline, it is more prone to errors when processing words at high speeds. However, it could help identify slow fingerspelling, which is common in early-stage SL learners with less signing fluency. The Bukva is thus intended to be (1) a pre-training source for a fingerspelling recognition model and (2) a training set for the baseline.

Models can produce inaccurate results when multiple people appear in the same frame, as Bukva's videos contain only one person. Finally, collecting a racially and ethnically diverse RSL dataset presents significant challenges, as the majority of deaf participants and RSL interpreters from crowdsourcing platforms do not represent ethnic minorities. As a result, one of the limitations of the presented dataset is the lack of racial heterogeneity.

Ethical Considerations

All crowdworkers recording the dataset signed informed consent for data processing and publication. We also ensured that the part of the Slovo dataset (Kapitanov et al., 2023) included in Bukva adheres to the described ethical requirements. We do not restrict videos with signers under 18 since parental permission was obtained during the registration, which complies with the Civil Code of the Russian Federation[6]. To preserve contributors' privacy, we employ anonymized user hash IDs in the dataset annotations. We provide the dataset for research purposes only. However, we recognize that it could be misused for malicious purposes, such as identifying individuals or enabling large-scale surveillance.

Positional Statement Throughout the research, we involved experts from the "All-Russian Society of the Deaf" and professional sign language interpreters. The expertise of the "All-Russian Society of the Deaf" (VOG) was utilized at every stage of the Bukva dataset creation. We also engaged deaf consultants in the development of training strategies to apply considerations to specific solutions. Additionally, some of our researchers completed formal courses on RSL to enhance their knowledge in this domain.

References

Mansour Alsulaiman, Mohammed Faisal, Mohamed Mekhtiche, Mohamed Bencherif, Tariq Alrayes, Ghulam Muhammad, Hassan Mathkour, Wadood Abdul, Yousef Alohali, Mansour Alqahtani, Habib Al-Habib, Hassan Alhalafi, Mohammed Algabri, Muneer Alhammadi, Hamdi Altaheri, and Taha Alfakih. 2023. Facilitating the communication with deaf people: Building a largest saudi sign language dataset. *Journal of King Saud University - Computer and Information Sciences*, 35(8):101642.

Andre L. C. Barczak, Napoleon H. Reyes, Mary E Abastillas, A. Piccio, and Teo Susnjak. 2011. A new 2d static hand gesture colour image dataset for asl gestures.

Danielle Bragg, Abraham Glasser, Fyodor Minakov, Naomi Caselli, and William Thies. 2022. Exploring collection of sign language videos through crowdsourcing. *Proceedings of the ACM on Human-Computer Interaction*, 6(CSCW2):1–24.

Ashley Chow, Glenn Cameron, Manfred Georg, Mark Sherwood, Phil Culliton, Sam Sepah, Sohier Dane, and Thad Starner. 2023. Google – american sign language fingerspelling recognition. Kaggle.

Philippe Dreuw, Thomas Deselaers, Daniel Keysers, and Hermann Ney. 2006. Modeling image variability in appearance-based gesture recognition. In *ECCV Workshop on Statistical Methods in Multi-Image and Video Processing*, pages 7–18.

M. G. Grif and Y. K. Kondratenko. 2021. Development of a software module for recognizing the fingerspelling of the russian sign language based on lstm. *Journal of Physics: Conference Series*, (1).

Jamaladdin Hasanov, Nigar Alishzade, Aykhan Nazimzade, Samir Dadashzade, and Toghrul Tahirov. 2023. Development of a hybrid word recognition system and dataset for the azerbaijani sign language dactyl alphabet. *Speech Communication*, 153:102960.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep residual learning for image recognition. *Preprint*, arXiv:1512.03385.

Hana Hosoe, Shinji Sako, and Bogdan Kwolek. 2017. Recognition of jsl finger spelling using convolutional neural networks. In *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, pages 85–88.

Byeongkeun Kang, Subarna Tripathi, and Truong Q. Nguyen. 2015. Real-time sign language fingerspelling recognition using convolutional neural networks from depth map. *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 136–140.

Alexander Kapitanov, Kvanchiani Karina, Alexander Nagaev, and Petrova Elizaveta. 2023. Slovo: Russian sign language dataset. In *Computer Vision Systems: 14th International Conference, ICVS 2023, Vienna, Austria, September 27–29, 2023, Proceedings*, page 63–73.

Alexander Kapitanov, Karina Kvanchiani, Alexander Nagaev, Roman Kraynov, and Andrei Makhliarchuk. 2024. Hagrid – hand gesture recognition image dataset. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4572–4581.

Chingiz Kenshimov, Samat Mukhanov, Timur Merembayev, and Didar Yedilkhan. 2021. A comparison of convolutional neural networks for kazakh sign language recognition. *Eastern-European Journal of Enterprise Technologies*, pages 44–54.

Taehwan Kim, Jonathan Keane, Weiran Wang, Hao Tang, Jason Riggle, Gregory Shakhnarovich, Diane Brentari, and Karen Livescu. 2016. Lexicon-free fingerspelling recognition from video: Data, models, and signer adaptation. *Comput. Speech Lang.*, 46:209–232.

Maksim Kuprashevich and Irina Tolstykh. 2023. Mivolo: Multi-input transformer for age and gender estimation. *Preprint*, arXiv:2307.04616.

[6] https://ihl-databases.icrc.org/en/national-practice/federal-law-no-152-fz-personal-data-2006

Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. 2020. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1459–1469.

Ji Lin, Chuang Gan, and Song Han. 2019. Tsm: Temporal shift module for efficient video understanding. *Preprint*, arXiv:1811.08383.

Ilya Loshchilov and Frank Hutter. 2017. Sgdr: Stochastic gradient descent with warm restarts. *Preprint*, arXiv:1608.03983.

Ilya Makarov, Nikolay Veldyaykin, Maxim Chertkov, and Aleksei Pokoev. 2019. Russian sign language dactyl recognition. In *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*, pages 726–729.

Jakub Piskozub. 2023. Letters of polish sign language alphabet.

Kanchon Kanti Podder, Muhammad Chowdhury, Zaid Mahbub, and Muhammad Kadir. 2020. Bangla sign language alphabet recognition using transfer learning based convolutional neural network. *The Bangladesh journal of scientific research*, 31-33:20–26.

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2019. Mobilenetv2: Inverted residuals and linear bottlenecks. *Preprint*, arXiv:1801.04381.

Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2019. Fingerspelling recognition in the wild with iterative visual attention. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5399–5408.

Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Jonathan Michaux, Diane Brentari, Gregory Shakhnarovich, and Karen Livescu. 2018. American sign language fingerspelling recognition in the wild. *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 145–152.

Ala Addin I. Sidig, Hamzah Luqman, Sabri Mahmoud, and Mohamed Mohandes. 2021. Karsl: Arabic sign language database.

Jhuma Sunuwar, Samarjeet Borah, and Aditi Kharga. 2023. Nsl23 dataset for alphabets of nepali sign language. SSRN.

Ewa Szyszka. 2021. Polish Sign Language Alphabet 100.

Pavan Kumar Anasosalu Vasu, James Gabriel, Jeff Zhu, Oncel Tuzel, and Anurag Ranjan. 2023. Mobileone: An improved one millisecond mobile backbone. *Preprint*, arXiv:2206.04040.

Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. 2023. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, and Matthias Grundmann. 2020. Mediapipe hands: On-device real-time hand tracking. *ArXiv*, abs/2006.10214.

A Dataset Creation Details

Since the RSL alphabet contains dynamic signs for 8 letters and static for others, we collect videos instead of images for all 33 letters to ensure clarity. We base the data creation pipeline on the one from Kapitanov et al. (2023). The pipeline consists of 3 main stages: (1) video collection, (2) video filtering and validation, and (3) sign time interval annotation. Bragg et al. (2022) has a nearly similar data creation pipeline and has several necessary steps, which include data recording, quality control, dataset review, and qualitative feedback. We utilized ABC Elementary[7] as a crowdsourcing platform through all stages. Note that we also employ another crowdsourcing platform, Yandex Toloka[8], through the first stage to diversify the group of signers and such context characteristics as background and lighting.

A.1 Exam

Since Russian sign language knowledge is infrequent, we conducted the qualifying stage to select crowdworkers to show RSL dactyl signs. We asked workers to watch 20 videos from the SpreadTheSign website[9] demonstrating RSL signs and choose the appropriate translations to spoken Russian language (see Figure 5). The exam time was limited to 10 minutes to avoid fake results from irresponsible people.

Since each stage requires specific RSL alphabet knowledge, the crowdworkers are required to pass a mandatory exam aimed at identifying natives and language learners. Figure 8 shows the dataset creation process. Hourly wages for assignments were comparable to or slightly higher than the national minimum wage in regions of the Russian Federation. The passed exam was an identifier for workers' access to video collection and validation stages. Crowdworkers who answered 80% of exam

[7] https://elementary.activebc.ru
[8] https://platform.toloka.ai/
[9] https://www.spreadthesign.com

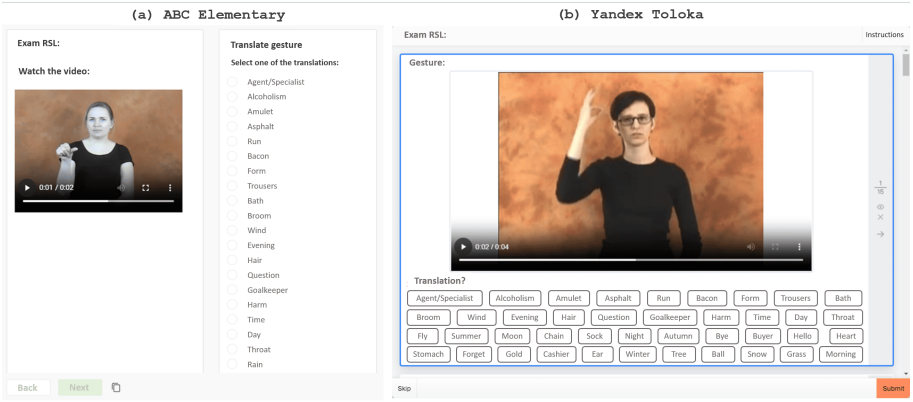


Figure 5: Example of a web page with an exam on crowdsourcing platforms translated into English.

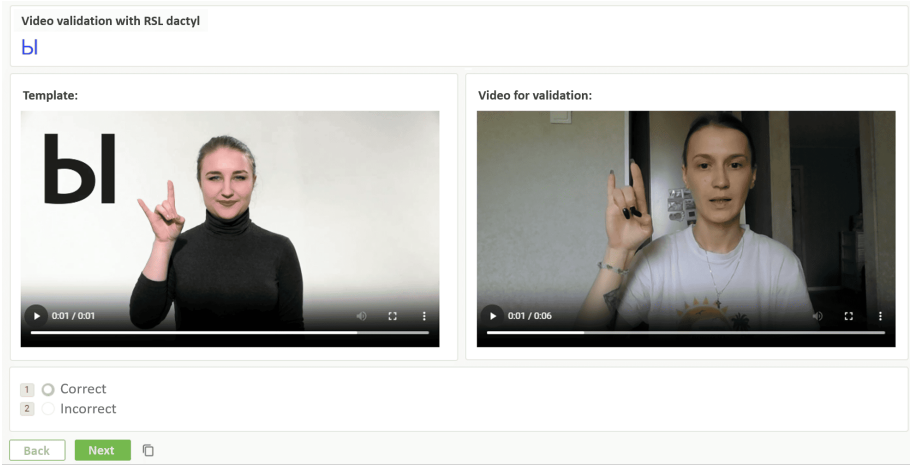


Figure 6: Example of a web page with a video validation on the ABC Elementary crowdsourcing platform translated into English.

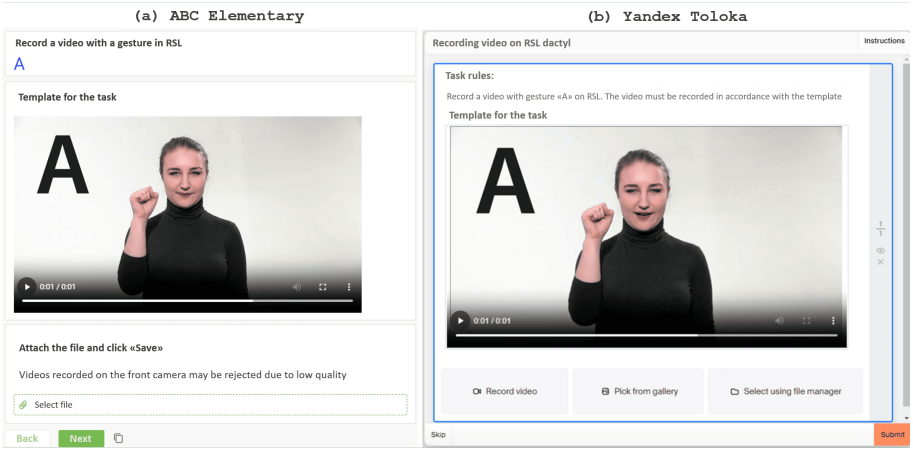


Figure 7: Example of a web page for recording videos on crowdsourcing platforms translated into English.



Figure 8: All crowdworkers are required to pass an exam before getting access to all crowd stages. During data collection, signers recorded the sign from the video template with further filtering to remove low-quality videos and duplicates. Only the videos remaining after the filtering stage were verified by at least three crowdworkers to ensure their correctness compared to the template. The annotation stage was also applied only to the samples remaining after the previous stage. Three annotators are asked to mark the video's sign start and end marks. Received marks are aggregated by the algorithm proposed in Kapitanov et al. (2023).

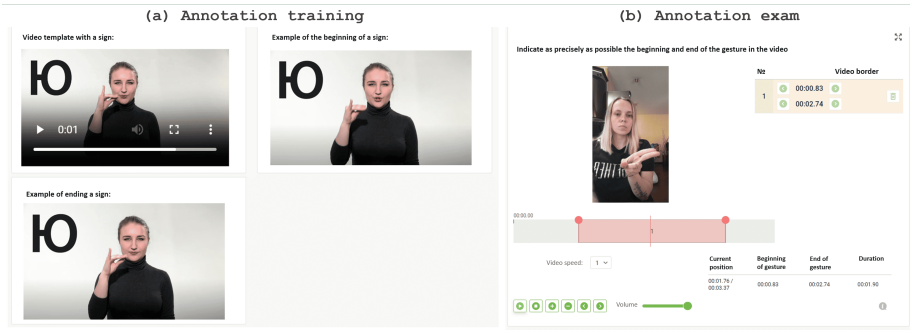


Figure 9: Example of a web page with a video annotation on the ABC crowdsourcing platform translated into English.

tasks correctly (at least 16 correct answers) were allowed to validate collected videos (see Figure 6), while access to the video recording stage required at most two mistakes (at least 18 correct answers).

Some participants were involved in crowdsourcing platforms from the "All-Russian Society of the Deaf" (VOG), while others were already crowdworkers. The data collection process included a survey on participants' level of deafness and reasons for knowing RSL. The results showed that about 60% are deaf, 30% are hard of hearing, and 10% are professional translators. It was discovered that approximately 20% of the participants are sign language learners, while the rest are fluent.

A.2 Video Collection

Crowdworkers were asked to show the RSL dactyl sign specified in the task and record the video through the crowdsourcing platform or load the video from the device's memory. A sign template from the YouTube channel[10] was provided for each letter of the RSL alphabet to simplify the process of recording the signs (see Figure 7). We attached instructions to this stage to outline the requirements: (1) only one clothed person is allowed in the frame, (2) the video should be of high quality and remain unshaken throughout, (3) editing or processing the video is prohibited, and (4) a hand performing a sign should be completely visible in the frame. We defined high-quality videos as those with a minimum frame rate of 15 frames per second (FPS), a minimum duration of 30 frames, and a resolution of at least 720 pixels on the smaller side. We have received approximately 8,400 videos at this stage, which will undergo further filtering and validation.

Compensation. We paid \$0.5 for each rule-abiding video, valuing the crowdworker's hour at \$8-10. It is important to note that the average price for common crowdworker tasks is 5 times less. Furthermore, professional sign language translators can earn up to \$10 per hour in this region.

A.3 Video Filtering and Validation

Filtering We left only videos with a frame's short edge of at least 720 pixels to subsequently receive the HD dataset. Also, we filtered out videos shorter than 30 frames and excluded those less than 15 frames per second (FPS). Further, videos were normalized to 30 FPS to align subsequent time-interval annotations. Since crowdsourcing platforms' usage is fraught with unscrupulous workers, the filtering stage was expanded with the deduplication process: crowdworkers may receive a task they have already completed and upload a previously recorded video. We identified duplicates by (1) checking all videos from the same user by user ID and (2) calculating the hash of some video keyframes for the entire dataset. To improve the final dataset's quality, we automatically excluded 286 (3.4%) low-quality videos. We also identified and rejected 439 (5.2%) duplicate videos during the deduplication process.

Validation The validation stage was added to the creation pipeline to check the correctness of sign demonstration. This stage can be considered an extra attempt to avoid wrong sign execution on par with the RSL alphabet exam. The workers were asked to compare the verifiable video with the sign template from the collection stage. Also, validators should check videos for the presence of hands entirely in the frame when a sign is being shown, the absence of other people, traces of processing, and watermarks. Note that including other individuals in the dataset is not feasible due to challenges in obtaining their consent. About 6,000 (71.4%) samples were approved after this stage.

To increase confidence in the decision, at least three different workers validate each video. In case of disagreement, another crowdworker is required to validate the video, allowing for a maximum of 5 reviewers. We accept samples with at least 70% positive votes. Note that validators are permitted to check only other people's videos.

Although the validators are not certified RSL instructors, several pipeline properties support the reliability of their judgments. First, only crowdworkers who passed the RSL proficiency exam with at least 80% correct answers were granted access to this stage. Second, validation is framed as a template-matching task: validators compare the verifiable video against the reference execution of the sign by a professional signer rather than provide a free-form expert judgment of sign correctness. Third, each decision aggregates the votes of three to five independent validators. Finally, as the survey described in Section A shows, fluent signers, most of whom are deaf or hard of hearing, constitute the majority of the crowdworker pool.

A.4 Time Interval Annotation

The process of sign recording frequently includes three parts: (1) the signer is preparing to perform

[10] https://www.youtube.com/watch?v=jtbwEalS0CE

the sign, (2) the signer shows the sign, and (3) the signer has already performed the sign (see Figure 8). Since only the second part is informative for the SLR model, the annotation stage was applied to separate the sign frames from the others. We asked the crowdworkers to indicate the time interval on the video with a sign by the sign's start and end marks (see Figure 9). Each video was annotated by three different workers with subsequent interval aggregation. We utilized the aggregation pipeline from Kapitanov et al. (2023) without modifications. Furthermore, videos were trimmed along the received marks.

A.5 Combination and Extra Filtering

We combined collected samples with 660 dactyl videos from the Slovo dataset to diversify classes on signers. The merged version was cleared of duplicates within a class. We left unique signers per class, receiving at least 101 samples in each.

Although validators checked videos for complete hand presence in the frame, some videos with the hand out of the frame leaked to the annotation stage. We added extra filtering to exclude such samples from the dataset completely. With a hand detector, we receive the bounding box annotations for hands in each video frame. Note that only the sign part[11] of the video is considered. We agreed that the hand is located out of the frame if there are 5 consecutive frames where more than half of the bounding box is outside the frame. The YOLOv7 (Wang et al., 2023) trained on the HaGRID (Kapitanov et al., 2024) dataset was utilized as a hand detector.

B Dataset Statistics

[11] Those, where the person shows the sign. At number 2 in Section A.4

Table 3: Dataset statistics. “Sign length” in frames, “Brightness” in grayscale levels, “Gender”, “Devices” by records, “Age” of crowdworkers and “Total” number of videos. The Latin transliteration of each Cyrillic letter is given in brackets.

Letters	**Sign length**			**Brightness**			**Gender**		**Devices**		**Age**			**Total**
	Min	Max	Median	Min	Max	Median	Male	Female	Smartphone	PC	Min	Max	Median	
А (A)	23	94	45.0	59	185	131	26	92	101	17	11.1	71.3	30.7	118
Б (B)	21	112	50.0	62	191	132	17	92	94	15	13.5	76.6	31.7	109
В (V)	22	105	44.0	57	192	129	21	95	95	21	10.4	71.7	32.4	116
Г (G)	25	100	43.0	69	190	125	19	92	94	18	12.7	63.9	31.4	112
Д (D)	32	105	50.0	65	190	130	20	82	84	18	11.4	66.1	31.1	102
Е (E)	22	101	47.0	66	186	129	22	97	99	20	10.8	74.4	31.8	119
Ё (Yo)	27	132	54.0	78	184	127	27	96	104	19	10.9	73.6	31.1	123
Ж (Zh)	22	125	48.5	31	206	123	28	90	99	19	12.0	77.6	29.6	118
З (Z)	28	92	51.0	67	191	127	24	84	92	16	11.8	75.3	31.9	108
И (I)	21	138	46.0	67	191	129	19	99	101	17	9.4	74.3	31.4	118
Й (Y)	25	111	48.0	65	194	129	22	85	89	18	8.7	75.0	31.7	107
К (K)	17	121	43.0	19	193	130	20	87	93	14	11.1	73.4	31.1	107
Л (L)	22	123	45.0	66	191	125	25	92	97	20	13.8	72.7	31.8	117
М (M)	24	140	46.5	63	177	125	21	83	89	15	10.0	72.1	31.1	104
Н (N)	28	124	47.0	54	189	128	22	96	99	19	14.2	65.5	31.5	118
О (O)	26	100	44.5	60	191	128	21	99	100	20	16.4	75.9	31.1	120
П (P)	22	124	47.0	58	190	126	22	96	95	23	10.8	71.9	33.0	118
Р (R)	23	90	50.0	60	191	128	22	93	98	17	11.7	74.2	30.6	115
С (S)	27	102	44.0	18	189	125	27	98	102	23	12.8	75.2	31.7	125
Т (T)	24	104	46.0	58	183	126	21	93	94	20	13.4	75.3	31.5	114
У (U)	24	182	45.0	31	193	129	21	92	98	15	10.1	62.5	30.4	113
Ф (F)	24	133	48.0	59	185	126	22	94	95	21	12.2	76.4	31.1	116
Х (Kh)	24	116	45.0	67	186	130	25	96	100	21	11.9	72.7	30.9	121
Ц (Ts)	24	85	47.0	51	200	128	24	86	93	19	12.5	74.6	31.8	112
Ч (Ch)	27	104	50.0	61	193	127	25	87	87	20	11.1	74.8	30.6	107
Ш (Sh)	28	132	49.0	18	194	129	23	96	100	19	12.2	73.3	30.5	119
Щ (Shch)	21	93	46.0	24	188	127	21	82	87	16	14.6	72.4	30.1	103
Ъ (hard sign)	26	124	48.0	55	191	128	26	89	97	18	13.9	81.2	32.2	115
Ы (Y)	23	174	43.0	73	185	127	24	99	105	18	11.7	72.9	31.9	123
Ь (soft sign)	24	124	44.0	21	187	131	18	83	90	11	12.3	64.0	31.7	101
Э (E)	22	156	47.0	22	191	129	24	86	92	18	9.8	73.6	30.9	110
Ю (Yu)	19	129	50.0	62	192	126	23	94	99	18	11.8	74.2	31.1	117
Я (Ya)	25	139	47.5	73	191	128	24	88	94	18	11.3	72.3	31.5	112

Cross-Lingual Transfer of Self-Supervised Representations for Sign Language Translation: A Case Study for Turkish Sign Language

Özge Can Özel[1*], Saliha Nur Şahin[1*], Murat Saraçlar[1]
[1]Boğaziçi University
ozge.ozel@std.bogazici.edu.tr saliha.sahin@std.bogazici.edu.tr murat.saraclar@bogazici.edu.tr
*Equal contribution

Abstract

Sign Language Translation (SLT) is especially difficult for low-resource languages such as Turkish Sign Language (TID), where aligned sign-text data are scarce. Self-supervised encoders pretrained on large sign corpora can reduce this dependence, but existing models are developed and evaluated almost exclusively on high-resource sign languages. One such model is SHuBERT (Sign Hidden-Unit BERT), a publicly released self-supervised contextual representation model trained on American Sign Language (ASL). We ask whether SHuBERT can be adapted to TID translation on the open-domain BUTID benchmark. Pretrained SHuBERT is incorporated into an encoder-decoder-based translation model and evaluated under both frozen and fine-tuned settings. This is compared against two references trained under the same recipe: a pose baseline, and a raw-feature setting using DINOv2-encoded hand/face crops and pose. All variants outperform the pose baseline, frozen SHuBERT performs on par with raw-feature setting, and fine-tuning yields the strongest result, reaching 7.15 BLEU-4 on the test set (+1.62 over the pose baseline, +0.77 over the raw-feature setting). Our results suggest that reusing a high-resource, ASL-pretrained self-supervised encoder can benefit low-resource SLT, but this benefit emerges through fine-tuning rather than frozen transfer alone.

1 Introduction

Sign languages convey meaning through complex visual patterns involving hand shapes, movements, facial expressions, and body posture. Sign Language Translation (SLT) remains challenging due to long-range spatio-temporal dependencies, and is especially difficult for low-resource and underrepresented sign languages such as Turkish Sign Language (TID), where aligned sign-text data are scarce.

Self-supervised pretraining on large unlabeled sign corpora has emerged as a way to reduce annotation requirements (Rust et al., 2024; Gueuwou et al., 2025). In particular, SHuBERT (Gueuwou et al., 2025), a transformer-based contextual representation model pretrained on 1,000 hours of ASL data, attains state-of-the-art results on ASL translation.

We study cross-lingual transfer of SHuBERT's ASL-pretrained representations to TID translation on the open-domain BUTID dataset (Şahin et al., 2025). We combine these representations across transformer layers via a learned weighted sum and train a linear projection together with an encoder-decoder-based translation model. The end-to-end translation model is evaluated under both a frozen-SHuBERT setting and a setting in which SHuBERT's own weights are fine-tuned jointly with the rest of the model.

To isolate where the improvement comes from, we compare these two settings against two references trained under the same recipe: a pose baseline using MediaPipe (Lugaresi et al., 2019) landmarks, and a raw-feature setting using SHuBERT's own four input streams (DINOv2-encoded (Oquab et al., 2024) hand/face crops and pose) without its transformer encoder.

Our contributions are:

- We evaluate cross-lingual transfer of SHuBERT, an ASL-pretrained self-supervised encoder, to TID translation.
- We compare frozen and fine-tuned configurations of pretrained SHuBERT against a raw-feature setting and a pose baseline.
- We analyze which SHuBERT transformer layers carry transferable information via their learned mixing weights.

Proceedings of the Second Workshop on Sign Language Processing, pages 162–171
October 29, 2026 ©2026 Association for Computational Linguistics

2 Related Work

Early SLT systems relied on gloss annotations (Camgöz et al., 2018), but the high cost of labeling has motivated a shift toward gloss-free, end-to-end approaches (Zhang et al., 2024). These systems benefit from large open-domain corpora such as YouTube-ASL (Uthus et al., 2023), but remain largely restricted to high-resource languages.

Self-supervised pretraining has recently been adapted to sign language videos. One such approach is SHuBERT (Gueuwou et al., 2025), which extends HuBERT (Hsu et al., 2021) (originally developed for speech, where it learns representations by a masked prediction paradigm) to a multi-stream sign representation, achieving state-of-the-art results on several ASL benchmarks. Because SHuBERT inherits HuBERT's transformer stack, it plausibly inherits a comparable layer-wise organization. In speech, layer-wise analyses of self-supervised models show that transformer layers encode qualitatively different information with depth, intermediate layers frequently carrying richer linguistic content than the final layers (Pasad et al., 2021). The standard downstream protocol aggregates all layers through learned weights rather than relying on any single one (Yang et al., 2021). We investigate whether analogous layer-specialization effects hold for SHuBERT when transferred to TID.

Prior TID resources focused largely on isolated sign recognition, the BosphorusSign corpus (Camgöz et al., 2016) and its curated BosphorusSign22k benchmark (Özdemir et al., 2020), alongside the large-scale AUTSL dataset (Sincan and Keles, 2020). Within TID, self-supervised features have so far been applied mainly to isolated recognition (Şero et al., 2025).

Continuous, sentence-level translation resources are comparatively recent, and work on TID translation remains limited. E-TSL (Öztürk and Keles, 2024) introduces a 24-hour educational-domain dataset with pose-based baselines, and BUTID (Şahin et al., 2025) provides the largest open-domain TID translation benchmark to date, which we adopt in this work. Even so, TID resources remain small relative to high-resource sign languages, motivating approaches that reuse representations learned elsewhere rather than training from scratch.

Cross-lingual transfer offers one such route: a translation model pretrained on a high-resource language benefits a low-resource one—for example, Alhejab et al. (2026) pretrain a T5 model on ASL and fine-tune on Saudi Sign Language, reporting a roughly threefold BLEU-4 gain over training from scratch. Closest in goal, Mo et al. (2025) pretrain a multi-stream skeleton masked autoencoder on YouTube-ASL and transfer it to several unseen sign languages for recognition and translation. Transfer has also been studied within TID: Kindiroglu et al. (2024) evaluate supervised cross-dataset transfer between two Turkish isolated SLR datasets (BosphorusSign22k and AUTSL) using a temporal graph-convolutional model.

Our work differs from all of these in two respects: (i) we transfer a publicly released self-supervised encoder (SHuBERT) *cross-lingually*, from high-resource ASL; and (ii) we target continuous TID translation, a setting underrepresented in prior work, rather than isolated recognition.

3 Methods

3.1 Dataset

We use the BUTID dataset (Şahin et al., 2025), a large-scale open-domain Turkish Sign Language corpus of broadcast television content (drama, comedy, and historical series) from the "Engelsiz TRT" platform, spanning 27 signers. The official splits are partitioned at the clip level rather than the series, episode, or signer level. Different clips from the same series, and potentially the same episode or signer, can appear across train, dev, and test. We use its *multi-caption* subset of long-form, multi-sentence clips (on average 4 sentences per clip): 106,665 / 5,925 / 5,925 train / dev / test clips (287.7 / 16.1 / 16.1 hours), averaging $\sim$9.7 s and $\sim$22 text tokens per clip. [1]

3.2 SHuBERT-Based Translation Model

The translation pipeline is illustrated in Figure 1. We run a combined translation model, following the method used in the SHuBERT paper (Gueuwou et al., 2025) but substituting ByT5 (Xue et al., 2022) with a multilingual T5 model (mT5-small) (Xue et al., 2021). We extract four visual streams (left hand, right hand, face, upper-body pose), via MediaPipe (Lugaresi et al., 2019) landmarks and

[1] These split sizes and durations differ from those in the original BUTID paper (e.g., 5,717 / 5,712 dev / test clips); we use the version available at the time of writing. We release our exact train/dev/test splits as CSV files, with permission from the dataset creators, at https://drive.google.com/drive/folders/1XOAC0CKR3sS9vKnoFIfIuW_pj0m_yrkO?usp=sharing.

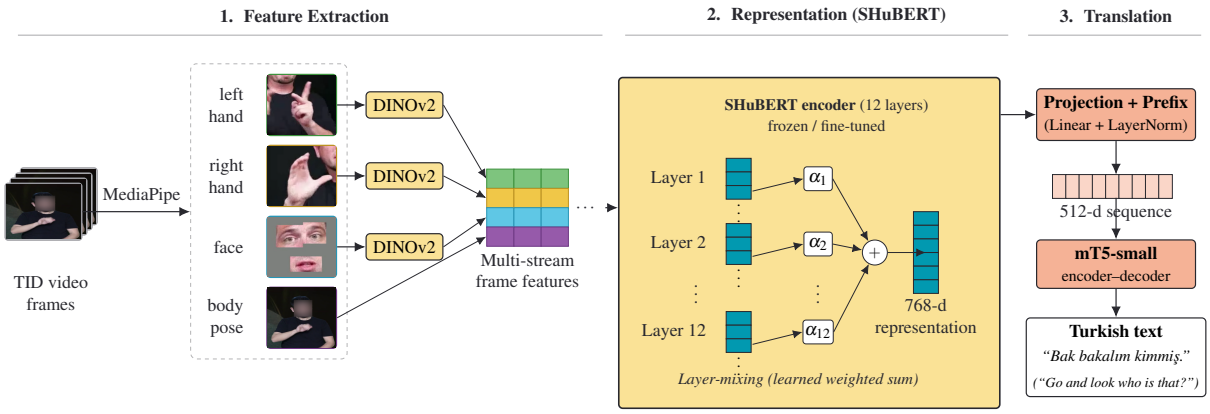


Figure 1: Overview of the SHuBERT-to-mT5 TID translation pipeline: four visual streams are encoded by SHuBERT, its per-layer 768-d states are combined by a learned weighted sum, projected into mT5-small, and decoded to Turkish.

fine-tuned DINOv2 models (Oquab et al., 2024)[2], for every BUTID clip. The four streams are then projected and concatenated to form a feature vector for each frame. The resulting sequence passes through SHuBERT's 12-layer transformer encoder (768-dimensional hidden states, 86M parameters), initialized from its public ASL-pretrained checkpoint[2]. We combine its per-layer hidden states with a learned weighted sum (Yang et al., 2021) rather than using a single layer's output. The result is projected through one linear layer into mT5-small's embedding space and prepended with an embedded task-prefix, then passed through mT5-small as a standard encoder-decoder, trained with token-level cross-entropy and a label smoothing factor of 0.2.

4 Experiments

We evaluate this pipeline under four conditions:

1. **Frozen SHuBERT:** SHuBERT is kept frozen; only the weighted-sum, projection, and mT5-small are trained.

2. **Fine-tuned SHuBERT:** SHuBERT is fine-tuned jointly with the rest of the model, warm-started from the converged frozen checkpoint in (1).

3. **Raw-feature:** To isolate SHuBERT's impact on translation performance, SHuBERT is removed entirely. The four raw streams are concatenated and projected directly into mT5-small's embedding space by a single linear layer. Through this, we isolate how much of the translation signal comes from SHuBERT's own representation as opposed to the raw visual features it is computed from.

4. **Pose baseline:** Pose-based baseline, trained under the same conditions as our other experiments, uses raw MediaPipe landmarks (8 body, 21 per hand, and 35 face points per frame). The landmarks are normalized relative to the hands and face, rescaled using a shared body-derived scale, flattened, and projected directly into mT5-small's embedding space through a single linear layer.

4.1 Training Configuration

All experiments were conducted on a single NVIDIA RTX 4090 GPU. Models are optimized with Adafactor (Shazeer and Stern, 2018), weight decay 0.01, a linear learning-rate schedule, batch size 4 with gradient accumulation over 64 steps (effective batch size 256), and the same label smoothing (0.2) throughout. Early stopping is performed based on validation performance. For all experiments except SHuBERT fine-tuning, we use a peak learning rate of 10^{-3} with a 2.4-epoch warmup and train for up to 100 epochs.

For fine-tuning, we use a discriminative learning rate: SHuBERT's own parameters receive a peak learning rate of 5×10^{-5} in a separate optimizer parameter group from the projection, layer-mixing, and mT5-small parameters. Rather than opening SHuBERT's weights from a random or

[2]We use the pretrained SHuBERT encoder and the face or hand fine-tuned DINOv2 feature extractors released in the official SHuBERT repository: https://github.com/ShesterG/SHuBERT.

purely frozen starting point, we warm-start fine-tuning from the best checkpoint of the corresponding frozen run, and open SHuBERT's encoder and layer-mixing weights simultaneously for continued training, with a shorter, 1-epoch warmup and a peak rate of 10^{-4} for the non-SHuBERT parameters.

Appendix A reports, for every system in Tables 1–3, the number of epochs run to the reported (best-dev) checkpoint and the corresponding optimizer steps.

4.2 Evaluation Metrics

To assess the quality of sign language translations, we employ standard evaluation metrics: BLEU (Papineni et al., 2002), ROUGE-L (Lin, 2004) and BERTScore (Zhang et al., 2020).

We report BLEU-1 through BLEU-4 to show how performance changes with the length of the matched n-gram. For a candidate translation and reference translation, BLEU combines n-gram precision with a brevity penalty. [3]

We additionally report ROUGE-L, which measures the longest common subsequence (LCS) between a hypothesis and its reference and converts the resulting precision and recall into an F-measure (Lin, 2004). [4]

Finally, we compute BERTScore (Zhang et al., 2020), which matches the most similar token in the other sentence using cosine similarity and aggregates these matches into precision, recall, and F_1.[5]

Several of the BLEU-4 margins we report are under 1 point, we therefore test the pairwise comparisons discussed in Section 5 for significance using paired bootstrap resampling (2000 resamples per test) on BLEU-4, via SacreBLEU's `--paired-bs` implementation (Post, 2018; Koehn, 2004).

4.3 Decoding Configuration

At inference, we decode with beam search (4 beams) and a length penalty of 1.0, blocking repetition of any 3-gram, and allow up to 128 new tokens with early stopping at the end-of-sequence token. We use the same decoding configuration for all systems, on both the development and test sets.

5 Results and Discussion

Table 1 reports results at each configuration's best checkpoint by dev BLEU-4. Both ASL-pretrained SHuBERT-based variants outperform pose baseline on every metric. Within the SHuBERT-based variants, the frozen weighted-sum model performs comparably to the raw-feature setting. Fine-tuning SHuBERT's own weights yields the largest gain observed, surpassing both the frozen and raw-feature configurations, with consistent trends between dev and test.

Paired bootstrap resampling on the test set confirms that fine-tuned SHuBERT significantly outperforms both the raw-feature and frozen configurations ($p < 0.001$ for both), while the difference between frozen SHuBERT and the raw-feature setting is not significant ($p = 0.354$). The pose baseline is also significantly below the raw-feature, frozen, and fine-tuned SHuBERT configurations ($p < 0.001$ for all three comparisons).

The absence of a significant difference between the frozen SHuBERT model and the raw-feature setting suggests that the pretrained representations may not transfer optimally to TID without adaptation.

One plausible explanation is that SHuBERT's pretrained encoder compresses its input toward representations useful for ASL. Under frozen transfer to a typologically distant language such as TID, this compression may discard information relevant to the target language, offsetting the benefit of useful contextual structure. This would also explain why fine-tuning, which allows SHuBERT's representations to adapt away from ASL-specific structure, recovers a clear advantage over both frozen configurations.

A translation baseline for the BUTID multi-caption subset is reported in the official dataset paper (Şahin et al., 2025): BLEU-1 37.61, BLEU-4 2.42, ROUGE-L 22.16, and BERTScore 50.96 on the test set. The two setups differ in several ways: our models use a single linear projection rather than a two-layer MLP sign encoder, the training configurations differ, and, most importantly, the evaluation metrics are computed using different procedures. We therefore treat the reported BUTID scores as context rather than a directly comparable baseline, and base our conclusions on the systems

[3] The implementation is based on SacreBLEU (Post, 2018) with signature `BLEU|nrefs:1|case:mixed|eff:no|tok:13a|smooth:exp|version:1.4.2`.

[4] ROUGE-L is computed using the Python `rouge` package: https://github.com/pltrdy/rouge.

[5] Computed with Hugging Face `evaluate` using `dbmdz/bert-base-turkish-cased`: https://huggingface.co/dbmdz/bert-base-turkish-cased. BERTScore is computed without baseline rescaling.

Method	Development Set						Test Set					
	B-1	B-2	B-3	B-4	R-L	BERT	B-1	B-2	B-3	B-4	R-L	BERT
Pose Baseline	28.85	15.30	9.13	5.75	17.25	58.81	28.73	15.11	8.94	5.53	16.95	58.63
DINOv2+Pose (raw-feature)	30.60	16.76	10.26	6.57	18.83	59.39	30.42	16.41	9.97	6.38	18.28	59.18
Frozen SHuBERT	29.87	16.40	10.10	6.53	18.51	59.38	29.95	16.30	9.98	6.42	18.60	59.34
Fine-tuned SHuBERT	**31.71**	**17.89**	**11.21**	**7.35**	**20.00**	**60.45**	**31.57**	**17.68**	**10.99**	**7.15**	**19.81**	**60.26**

Table 1: SLT results on the BUTID development and test sets (Multi-Caption split). B-1 to B-4 denote BLEU-1 to BLEU-4, R-L is ROUGE-L, and BERT is the mean F1 of BERTScore.

Init.	**Adapt.**	**B-1**	**B-4**	**R-L**	**BERT**
Random	Frozen	25.87	3.81	12.62	54.93
Random	Fine-tuned	26.93	4.40	14.04	55.92
Pretrained	Frozen	29.95	6.42	18.60	59.34
Pretrained	Fine-tuned	**31.57**	**7.15**	**19.81**	**60.26**

Table 2: Impact of ASL pretraining on SHuBERT-based SLT (BUTID test set; checkpoints selected on dev BLEU-4, dev results follow the same trend). All variants share the same architecture; only the encoder initialization (*random* vs. pretrained SHuBERT) and adaptation differ.

we evaluate under a single fixed configuration.

5.1 Impact of ASL pretraining

In order to determine whether SHuBERT's gains arise from ASL pretraining itself or merely from the added encoder capacity, we evaluate two additional variants. Instead of beginning with the ASL-pretrained SHuBERT encoder, weights are randomly initialized. The same warm-started fine-tuning protocol is applied to the randomly-initialized variant. All experiments share the same architecture and parameter count (Section 4.1).[6] Results are reported in Table 2.

In the frozen setting, ASL pretraining raises BLEU-1 from 25.87 to 29.95 and ROUGE-L from 12.62 to 18.60 without updating any encoder parameters, indicating that the pretrained features themselves encode transferable signing structure. All pairwise differences in Table 2 are statistically significant ($p < 0.001$, paired bootstrap resampling). This gap persists after fine-tuning: the ASL-pretrained encoder leads by 4.6 BLEU-1 and 5.8 ROUGE-L over its randomly initialized counterpart, and the frozen ASL-pretrained encoder alone already surpasses the fully fine-tuned random one. While suggestive of a genuine pretraining benefit beyond added capacity, this comparison is confounded by the training protocol, and we cannot fully rule out that the randomly initialized model's fine-tuned performance is limited by optimization difficulty rather than the absence of ASL pretraining.

[6]We also attempted training both the pretrained and the randomly initialized encoder jointly with mT5-small end-to-end under a standard 10^{-3} schedule, without warm-starting from a frozen checkpoint; in both cases the training resulted in models with poor performance (BLEU-4 remained below 1.0), motivating our staged protocol.

5.2 Layer Importance Analysis

The layer-weighted-sum mechanism learns a softmax-normalized importance weight for each SHuBERT layer. We inspect these weights at the best checkpoint of all four weighted-sum configurations. Figures 2 and 3 compare the learned layer-weight distributions for randomly initialized and ASL-pretrained SHuBERT in the frozen and fine-tuned settings, respectively.

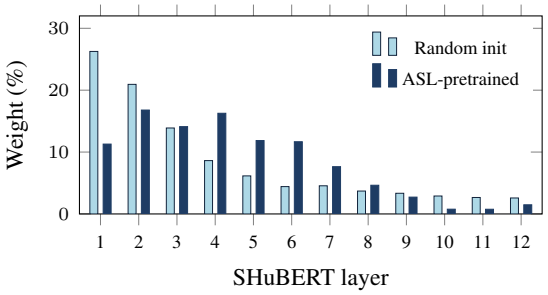


Figure 2: Layer-mixing weights, frozen setting: random initialization vs. ASL-pretrained SHuBERT.

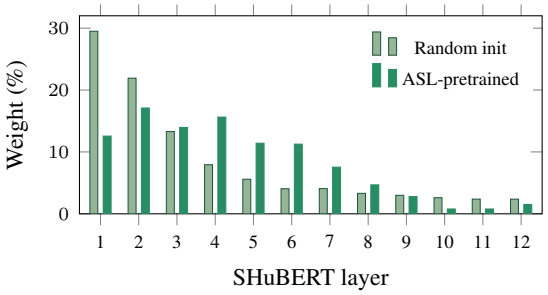


Figure 3: Layer-mixing weights, fine-tuned setting: random initialization vs. ASL-pretrained SHuBERT.

Method	B-1	B-4	R-L	BERT
Frozen (12 layers)	29.95	6.42	18.60	59.34
Frozen (Layer 4)	29.38	6.12	17.69	58.72

Table 3: Comparison of frozen SHuBERT using all layers and Layer 4 only (BUTID test set; checkpoints selected on dev BLEU-4, dev results follow the same trend).

ASL pretraining changes which layers the model relies on. Without pretraining, the weights concentrate sharply on the earliest layers (layers 1–2 alone carry 47–51% of the mass) and decrease steadily with depth. With ASL pretraining, the distribution is flatter: the model draws on the middle layers (roughly 1–5) fairly evenly before tailing off after layer 6.

The distribution remains nearly unchanged after fine-tuning for both settings, suggesting the frozen model already identifies the transferable layers, with fine-tuning refining representations rather than reweighting them. We note, however, that our fine-tuning runs are warm-started from a converged frozen checkpoint and trained for relatively few additional epochs (Section 4.1), so we cannot rule out that longer or non-warm-started fine-tuning would eventually shift the mixing weights further.

This concentration of weight in the early-to-intermediate layers raises the question of whether a single highly weighted layer can capture most of the benefit of the full weighted-sum representation. To examine this, we additionally evaluate a frozen model using only Layer 4, which individually receives one of the highest mixing weights ($\sim$16%) in the frozen and pretrained SHuBERT setting. For this experiment, the projection and the decoder are retrained from scratch under the same frozen protocol as the full weighted-sum model; only the layer-mixing is replaced by a fixed selection of Layer 4. As seen in Table 3, the Layer 4-only model achieves performance comparable to the full 12-layer frozen representation: the difference is statistically significant ($p = 0.004$, paired bootstrap resampling) but the absolute margin is small (0.30 BLEU-4). Isolating Layer 4 recovers most of the weighted-sum's benefit, indicating that the transferable signal is concentrated in a few layers rather than spread across the encoder.

5.3 Length-Stratified Analysis

Aggregate BLEU averages over utterances of very different lengths, which can mask systematic differences. To examine this, we stratify the development set clips into five bins by reference word count and compute SacreBLEU corpus-level BLEU-4 separately for each bin (Table 4).

Length	# Sent.	n	Pose	Raw	Frozen	FT
2–8	2.23	1185	11.15	11.32	10.74	**12.24**
8–12	2.58	1184	7.31	8.00	7.47	**8.46**
12–17	3.29	1185	6.61	7.19	7.27	**7.77**
17–26	4.63	1184	5.44	6.10	6.42	**7.10**
26–71	7.44	1185	4.30	5.64	5.50	**6.27**

Table 4: Dev BLEU-4 by reference length, using five approximately equal-sized bins. # Sent.: average number of sentences per clip; n: references per bin; Raw: raw-feature; FT: fine-tuned SHuBERT.

Fine-tuned SHuBERT outperforms the pose baseline in every bin. Its margin remains similar across the first three bins ($+1.09$–$+1.16$ BLEU-4) and increases for longer clips, reaching $+1.97$ in the longest bin. The raw-feature and frozen settings also outperform the pose baseline in most bins, although the frozen model trails pose in the shortest bin. BLEU-4 declines with length for every system, consistent with the greater difficulty of translating longer, multi-sentence utterances. In summary, the results suggest that fine-tuned SHuBERT becomes advantageous over the pose baseline as utterance length grows.

5.4 Qualitative Analysis

Table 5 shows four illustrative test-set examples comparing the fine-tuned SHuBERT model against the pose baseline.

In Example 1, SHuBERT reproduces the reference almost word-for-word while the pose baseline shifts tense and drops a clause.

In Example 2, SHuBERT matches the reference exactly, whereas the pose baseline loses the temporal marker *"bugün"* (today) and flattens the follow-up question into a statement.

In Example 3, both models correctly preserve the proper noun "Elizabeth" but generate an incorrect name for the addressee. However, SHuBERT accurately reproduces both the capture event and the successful completion of the operation, whereas the pose baseline omits the capture event. The higher sentence-level BLEU-4 (40.46 vs. 8.18) is consistent with the stronger performance of fine-tuned SHuBERT on longer, multi-clause examples discussed in Section 5.3.

In Example 4, the pose baseline reproduces the

reference's opening exchange verbatim, including the numeral *"elli kere"* (fifty times), but neither system renders the idiom *"gözleri dolu dolu bakıyor"* (eyes welling up), each instead substituting an unrelated closing clause, suggesting this kind of non-literal, idiomatic expression remains difficult for both architectures.

6 Conclusion

We studied cross-lingual transfer of SHuBERT's ASL-pretrained representations to low-resource, underrepresented Turkish Sign Language translation on BUTID. ASL-pretrained SHuBERT consistently outperforms random initialization, indicating that its pretrained representations contain transferable information. However, frozen SHuBERT performs similarly to the raw-feature setting, while fine-tuning the SHuBERT encoder yields the strongest results, reaching 7.15 BLEU-4 on the test set.

Our analyses further show that the transferable information is concentrated primarily in early-to-intermediate SHuBERT transformer layers, with a single highly weighted intermediate layer recovering most of the performance of the full weighted-sum representation. Fine-tuned SHuBERT also maintains an advantage over the pose baseline across utterance lengths, with a larger margin on longer inputs. Together, these results suggest that reusing a high-resource, ASL-pretrained self-supervised encoder can benefit low-resource sign language translation, though this benefit emerges through fine-tuning rather than frozen transfer alone.

Limitations

Our study is limited to a single target sign language and dataset (TID/BUTID), so it remains unclear whether the observed transfer patterns generalize to other low-resource sign languages or domains. Computational constraints also limit our translation model to mT5-small and restrict the extent of SHuBERT fine-tuning; larger translation models may interact differently with the transferred representations. We rely on automatic metrics, which under-reflect semantic adequacy for morphologically rich, word-order-flexible Turkish and which we do not complement with human evaluation. Despite the improvements, absolute translation quality remains low on this open-domain benchmark for all systems.

All experiments are run with a single random seed; we do not report variance across training seeds, though we assess the significance of our main pairwise comparisons via bootstrap resampling on the test set.

We also do not evaluate transfer under more severely data-scarce conditions; such an experiment would more directly test whether SHuBERT's benefit grows as target-language supervision becomes scarcer, and we leave this to future work.

Acknowledgments

This work was supported in part by the Turkish Directorate of Strategy and Budget through the ROYAL Project under Grant 2019K12-149250. We also thank Prof. Dr. Lale Akarun and Karahan Şahin for their support and guidance, and the anonymous reviewers for their constructive feedback.

References

Ali Alhejab, Tomáš Železný, Lamya Alkanhal, Ivan Gruber, Yazeed Alharbi, Jakub Straka, Václav Javorek, Marek Hrúz, Badriah Alkalifah, and Ahmed Ali. 2026. Saudi sign language translation using T5. In *Speech and Computer*, pages 331–343, Cham. Springer Nature Switzerland.

Necati Cihan Camgöz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7784–7793.

Necati Cihan Camgöz, Ahmet Alp Kındıroğlu, Serpil Karabüklü, Meltem Kelepir, A. Sumru Özsoy, and Lale Akarun. 2016. BosphorusSign: A Turkish Sign Language recognition corpus in health and finance domains. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1383–1388, Portorož, Slovenia. European Language Resources Association (ELRA).

Shester Gueuwou, Xiaodan Du, Greg Shakhnarovich, Karen Livescu, and Alexander H. Liu. 2025. SHuBERT: Self-supervised sign language representation learning via multi-stream cluster prediction. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 28792–28810, Vienna, Austria. Association for Computational Linguistics.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

Example	System	Output (Turkish)	English gloss
1	Reference	Seni bir daha hayal kırıklığına uğratmayacağım. En azından kendi adıma söz veriyorum. Ben elimden geleni yapacağım.	I won't let you down again. At least I promise on my own behalf. I will do everything I can.
	Fine-tuned SHuBERT	Seni bir daha hayal kırıklığına uğratmayacağım. En azından benim adıma söz veriyorum. Elimden geleni yapacağım.	I won't let you down again. At least I promise on my behalf. I will do everything I can.
	Pose baseline	Seni hayal kırıklığına uğratamam. En azından ben kendim adıma söz verdim. Elimden gelen her şeyi yaptım.	I can't let you down. At least I promised on my own behalf. I did everything I could.
2	Reference	Bugün çok garip bir şey oldu biliyor musun? Ne oldu?	Do you know something really strange happened today? What happened?
	Fine-tuned SHuBERT	Bugün çok garip bir şey oldu biliyor musun? Ne oldu?	Do you know something really strange happened today? What happened?
	Pose baseline	Bir şey mi oldu biliyor musun? Oldu.	Do you know if something happened? It did.
3	Reference	Komutanım, Elizabeth'i ele geçirdik. Operasyon başarıyla tamamlandı. Bu harika bir haber Emre. Peki konuştu mu? Şimdilik hayır ama konuşacaktır.	Commander, we've captured Elizabeth. The operation was completed successfully. That's great news, Emre. Did she talk? Not yet, but she will.
	Fine-tuned SHuBERT	Komutanım, Elizabeth'i ele geçirdik. Operasyon başarıyla tamamlandı. Harika haber Elif. Bunu mu söylüyorsun? Burada değil ama söyleyeceğim.	Commander, we've captured Elizabeth. The operation was completed successfully. Great news, Elif. Is that what you're saying? Not here, but I'll tell you.
	Pose baseline	Komutanım, Elizabeth'in operasyonu başarıyla tamamlandı. Harika haber Lilith. Sen ne dediğinin farkında mısın? Hayır, konuşmayı tercih ederim.	Commander, Elizabeth's operation was completed successfully. Great news, Lilith. Are you aware of what you're saying? No, I'd rather talk.
4	Reference	Bir şey mi oldu? Bilmiyorum abla. Elli kere sordum. Bir tuhaf, sadece gözleri dolu dolu bakıyor.	Did something happen? I don't know, sis. I've asked fifty times. It's strange, she just stares with teary eyes.
	Fine-tuned SHuBERT	Bir şey mi oldu? Bilmiyorum abla en başından beri soruyorum. Bozma moralini. Sanki bütün gün susuyorum.	Did something happen? I don't know, sis, I've been asking since the very beginning. Don't get upset. It's like I've been quiet all day.
	Pose baseline	Bir şey mi oldu? Bilmiyorum abla. Elli kere sordum. Sanki biraz tatlı kokuyor.	Did something happen? I don't know, sis. I've asked fifty times. It's like it smells a little sweet.

Table 5: Qualitative examples from the BUTID test set, comparing the fine-tuned SHuBERT model against the pose baseline. English glosses are our own translations, provided for readability.

Ahmet Alp Kindiroglu, Ozgur Kara, Ogulcan Ozdemir, and Lale Akarun. 2024. Transfer learning for cross-dataset isolated sign language recognition in under-resourced datasets. In *2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG)*, pages 1–8.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. MediaPipe: A framework for perceiving and processing reality. In *Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR) 2019*.

Junwen Mo, MinhDuc Vo, and Hideki Nakayama. 2025. Improving sign language understanding with a multi-stream masked autoencoder trained on ASL videos. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 1202–1218, Mumbai, India. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, and 7 others. 2024. DINOv2: Learning robust visual features without supervision. *Trans-*

actions on Machine Learning Research. Featured Certification.

Oğulcan Özdemir, Ahmet Alp Kındıroğlu, Necati Cihan Camgöz, and Lale Akarun. 2020. Bosphorus-Sign22k sign language recognition dataset. In *Proceedings of the LREC2020 9th Workshop on the Representation and Processing of Sign Languages: Sign Language Resources in the Service of the Language Community, Technological Challenges and Application Perspectives*, pages 181–188, Marseille, France. European Language Resources Association (ELRA).

Şükrü Öztürk and Hacer Yalim Keles. 2024. E-TSL: A continuous educational Turkish sign language dataset with baseline methods. In *2024 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, pages 1–7.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Ankita Pasad, Ju-Chieh Chou, and Karen Livescu. 2021. Layer-wise analysis of a self-supervised speech representation model. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 914–921.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Phillip Rust, Bowen Shi, Skyler Wang, Necati Cihan Camgöz, and Jean Maillard. 2024. Towards privacy-aware sign language translation at scale. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8624–8641, Bangkok, Thailand. Association for Computational Linguistics.

Karahan Şahin, Lale Akarun, Kadir Gökgöz, and Murat Saraçlar. 2025. BUTID: A large-scale sign language translation dataset and benchmarks for Turkish sign language. Presented at the 3rd Workshop on Sign Language Recognition, Translation and Production (SLRTP), CVPR 2025. Remote paper.

Onur Şero, Oğulcan Özdemir, and Lale Akarun. 2025. Turkish sign language recognition through self-supervised learning features and heatmap. In *2025 33rd Signal Processing and Communications Applications Conference (SIU)*, pages 1–4.

Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive learning rates with sublinear memory cost. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4596–4604. PMLR.

Özge Mercanoglu Sincan and Hacer Yalim Keles. 2020. AUTSL: A large scale multi-modal Turkish sign language dataset and baseline methods. *IEEE Access*, 8:181340–181355.

Dave Uthus, Garrett Tanzer, and Manfred Georg. 2023. YouTube-ASL: A large-scale, open-domain American sign language-English parallel corpus. In *Advances in Neural Information Processing Systems*, volume 36, pages 29029–29047. Curran Associates, Inc.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y. Lin, Andy T. Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, Tzu-Hsien Huang, Wei-Cheng Tseng, Ko tik Lee, Da-Rong Liu, Zili Huang, Shuyan Dong, Shang-Wen Li, Shinji Watanabe, Abdelrahman Mohamed, and Hung yi Lee. 2021. SUPERB: Speech Processing Universal PERformance Benchmark. In *Interspeech 2021*, pages 1194–1198.

Biao Zhang, Garrett Tanzer, and Orhan Firat. 2024. Scaling sign language translation. In *Advances in Neural Information Processing Systems*, volume 37, pages 114018–114047. Curran Associates, Inc.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations (ICLR)*.

A Training Epochs and Compute Budget

All experiments use a single fixed random seed (42). With batch size 4 and 64-step gradient accumulation, one optimizer step aggregates 256 clips, so each epoch over the 106,665-clip train split (Section 3.1) corresponds to ≈417 optimizer steps.

Table 6 reports, for every system in Tables 1–3, the number of training epochs completed up to the checkpoint selected by best dev BLEU-4 (Section 4.2), and the corresponding number of optimizer steps (each aggregating 64 microbatches of batch size 4, per Section 4.1). The frozen and

fine-tuned SHuBERT variants (and their random-init counterparts) are two-stage: fine-tuning warm-starts from the corresponding frozen run's own best checkpoint (Section 4.1), so we report the fine-tuning phase's epochs/steps as an addition on top of its frozen phase's own cost.

System	Epochs	Steps
Pose Baseline	66	27,500
Raw-feature	33	13,750
Frozen SHuBERT	38	15,833
+ fine-tuning	+8	+3,333
Frozen, layer 4	51	21,250
Random/Frozen	40	16,666
+ fine-tuning	+8	+3,333

Table 6: Epochs and optimizer steps to the reported (best-dev) checkpoint, per system. Rows marked "+ fine-tuning" are additional cost on top of the frozen row directly above them.

SpaMo-BdSL: Gloss-Free Bangla Sign Language Translation Using Spatial Configurations and Motion Dynamics with LLMs

Kowshik Saha[1], Adibah Mahmud[2], and M Saifur Rahman[3]
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology

[1]kowshiksahakabya@gmail.com, [2]adibahmahima@gmail.com, [3]mrahman@cse.buet.ac.bd

Abstract

Sign language translation (SLT) can help bridge the communication gap between the Deaf and hard-of-hearing community and the general population. Yet progress has been driven largely by a handful of high-resource sign languages, such as German or American Sign Language, supported by large annotated corpora and large computing budgets. Many sign languages, including Bangla Sign Language (BdSL), remain underrepresented because annotated data is scarce and much research assumes access to powerful hardware. In this paper, we share an early-stage adaptation of the gloss-free SpaMo framework for Bangla, called **SpaMo-BdSL**, and reflect on what we learned while building it under real resource limits. We report two findings that we believe matter for low-resource and underrepresented SLT. First, a small language-specific model (BanglaT5, 247M parameters) matches the translation quality of a much larger multilingual model (mT0-XL, 3.7B parameters) on our Bangla dataset, while being far cheaper to run; for example, it translates the test set in 40 seconds versus 2 minutes 43 seconds for mT0-XL. Second, a set of simple training changes suggested by a large language model (Claude Opus 4.6) allowed us to train the whole system on a single consumer-grade GPU. We argue that the common assumption "bigger and more multilingual is better" can quietly exclude underrepresented sign languages and the researchers who work on them, and we use our experience to open a discussion on more inclusive, multilingual, and culturally aware SLT.

1 Introduction

Sign language is the main way many Deaf and hard-of-hearing people communicate. It is visual and uses hand shapes, facial expressions, body posture, and movement instead of sound. Systems that automatically translate sign language videos into written text can help Deaf and hearing people understand each other. This is an accessibility problem at heart, and it is central to building inclusive, multilingual, and culturally aware SLT that serves underrepresented sign-language communities, not only the few languages that already have large resources.

Most SLT work carries two hidden assumptions. The first is about *language*: most systems target high-resource sign languages such as German Sign Language (Camgoz et al., 2018; Cihan Camgöz et al., 2020) or American Sign Language, and often rely on *gloss* annotations (a written label per sign) that are costly to produce and do not exist for many languages. The second is about *compute*: strong results are usually reported with large models and large GPUs that many researchers cannot access. Both assumptions can quietly exclude low-resource languages like Bangla Sign Language (BdSL) and the people who study them.

Recent gloss-free methods translate directly from video to text and remove the need for glosses (Zhou et al., 2023). One such method, SpaMo (Hwang et al., 2025), uses frozen off-the-shelf visual encoders to capture spatial configurations and motion dynamics, then feeds these features to a large language model (LLM). This design is attractive for low-resource work because it avoids expensive fine-tuning of the visual encoders.

We adapt SpaMo to Bangla, calling it **SpaMo-BdSL**, and report what happened when we ran it within our own limits. Our contributions are:

- We adapt the gloss-free SpaMo framework to BdSL by running the BanglaGov dataset (BanglaGov, 2025) through its pipeline.
- We compare a small Bangla-specific model (BanglaT5 (Bhattacharjee et al., 2023), 247M) with a much larger multilingual model (mT0-XL (Muennighoff et al., 2023), 3.7B) on the same data, and find that the small model

Proceedings of the Second Workshop on Sign Language Processing, pages 172–176
October 29, 2026 ©2026 Association for Computational Linguistics

matches the larger one's accuracy while being much cheaper and faster.

- We document concrete training changes, suggested by Claude Opus 4.6, that let us train the full pipeline on a single consumer-grade GPU, and discuss why each helps.

We connect these findings to a central question for underrepresented sign languages: how can SLT progress reach languages and communities that lack large annotated corpora and large compute, rather than only those already well served?

2 The SpaMo-BdSL Framework

Our system follows SpaMo (Hwang et al., 2025) with adaptations for Bangla. Figure 1 shows the pipeline. It has three parts: (I) sign feature extraction, (II) visual–textual alignment, and (III) text generation with a language model.

Feature extraction. We keep the two visual encoders frozen. A CLIP ViT-L/14 (Radford et al., 2021) is the *spatial* encoder, producing (with multi-scale S2 processing) one spatial vector per frame; a VideoMAE-L/16 (Tong et al., 2022) is the *motion* encoder, producing one motion vector per 16-frame clip via a sliding window. Because the encoders are frozen, we extract all features once and store them, saving a lot of compute during training.

Fusion and language model. The spatial and motion features are projected to a shared size, joined along time, and passed through a temporal convolutional *Sign Adapter* that shortens the sequence and captures local timing. A small MLP maps the result to the language model's embedding size. We then add a Bangla prompt ("ইশারা ভাষা থেকে বাংলায় অনুবাদ করুন:", "Translate from sign language to Bangla:") and feed everything to the language model, which generates the sentence. We fine-tune it with LoRA (Hu et al., 2021) (rank 16, $\alpha = 32$, dropout 0.1), so only a small number of parameters are trained.

Training objective. We combine the standard cross-entropy loss for text generation with a symmetric CLIP-style contrastive loss (Radford et al., 2021) that pulls matching sign–text pairs closer in the embedding space, giving $\mathcal{L} = \mathcal{L}_{\text{CE}} + \alpha\,\mathcal{L}_{\text{CLIP}}$ with $\alpha = 1.0$. The contrastive term helps in the gloss-free setting because there are no intermediate sign labels to supervise the visual features.

Dataset. We use the BanglaGov dataset (BanglaGov, 2025): 1,922 unique sentences, each signed by 5 signers, for 9,610 videos in total. The videos are high quality with clean backgrounds, and the domain is narrow (eating-related activities), so vocabulary and sentence patterns repeat often. We split the data 80%/10%/10% (train/dev/test) by sentence ID, so all five signers of a sentence fall in the same split. A sentence therefore never appears in more than one split, which prevents data leakage.

3 Adapting to a consumer-grade GPU

The original SpaMo settings assume a data-center GPU, and on our single consumer-grade GPU the defaults ran out of memory. We used one NVIDIA RTX 5060 Ti GPU with 16GB VRAM. We described our limits to the Claude Opus 4.6 language model, which suggested alternative settings for each bottleneck; we then tried these ourselves and kept the combination that actually ran. We report this openly because such human–AI collaboration is part of how inclusive tools get built, and because the reasoning is simple and reusable.

Table 1 lists the changes:

- **Feature-extraction batch size** ($32 \rightarrow 8$) for the ViT and VideoMAE stages. Multi-scale (S2) processing runs each frame through the ViT as five 224×224 tiles, and each VideoMAE "batch" is 16-frame clips, so memory grows quickly. Dropping the batch to 8 cuts peak memory by roughly $4\times$. Since the encoders are frozen and run only once, this costs a little extra time but leaves the extracted features unchanged.

- **Maximum frame length** ($512 \rightarrow 256$)**.** Videos longer than the cap are randomly cropped to a contiguous window. We use 256 because it is the largest cap that fits in memory, and it matches the data well: a 256-frame window fully covers 87.46%/87.19%/83.94% of BanglaGov train/dev/test videos and 98.94%/99.81%/100.00% of PHOENIX14T (Camgoz et al., 2018). Because the crop is random, the few cropped videos are seen from different windows across epochs, acting a little like augmentation.

- **Training batch size** ($8 \rightarrow 2$) **with gradient accumulation** ($2 \rightarrow 8$)**.** The *effective* batch size is unchanged at $8 \times 2 = 2 \times 8 = 16$, so

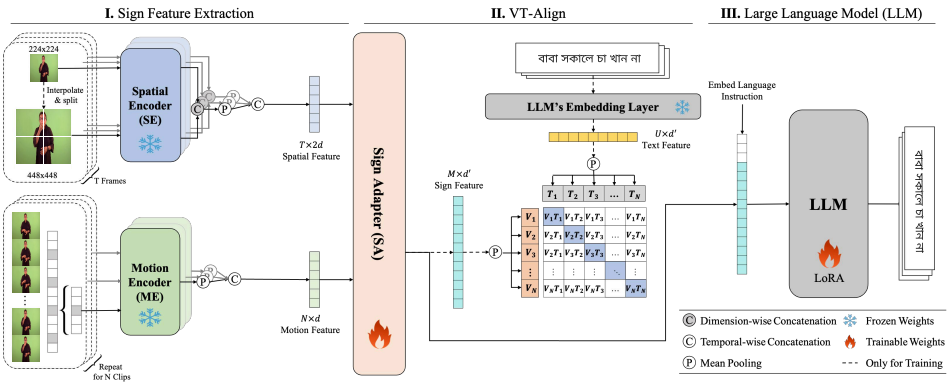


Figure 1: Overview of the SpaMo-BdSL pipeline. A frozen CLIP ViT-L/14 extracts spatial features and a frozen VideoMAE-L/16 extracts motion features; these are projected, fused by a temporal convolutional Sign Adapter, and passed with a Bangla prompt to a LoRA-tuned text model that generates the sentence. A contrastive loss (VT-Align) aligns the visual and text spaces. (Adapted from Hwang et al. (Hwang et al., 2025).)

Setting	Original	Ours
ViT extraction batch size	32	8
VideoMAE extraction batch size	32	8
Max. frame length	512	256
Training batch size	8	2
Gradient accumulation steps	2	8
Max. epochs (cap)	500	200

Table 1: Configuration changes to fit a single consumer-grade GPU. The effective training batch size is unchanged (16).

optimization behaves like the original while far fewer samples sit in memory at once.

- **Maximum epochs** ($500 \rightarrow 200$)**.** This is only a safety ceiling: we validate every 2 epochs and stop early if BLEU-4 does not improve for 50 checks. Since the original was trained for 40 epochs, 200 is a comfortable cap that also bounds wall-clock time.

None of these changes require new algorithms or extra data; they are careful bookkeeping of memory. This matters for inclusion: reproducing assistive-technology research should not require a data center. Running the full pipeline on our machine took a total of four days.

4 Small Language-Specific vs. Large Multilingual Models

A common instinct is to reach for the biggest, most multilingual model available. Bangla uses a non-Latin script, so a natural choice is a large multilingual model that already covers many scripts, such as mT0-XL (Muennighoff et al., 2023) (3.7B parameters). We compare it against BanglaT5 (Bhattacharjee et al., 2023) (247M parameters), a much smaller model pre-trained only on Bangla. Both are dropped into the exact same SpaMo-BdSL pipeline; only the language model changes.

Metric	mT0-XL (3.7B)	BanglaT5 (247M)
BLEU-1	96.041	**96.301**
BLEU-2	93.741	**93.792**
BLEU-3	**91.782**	91.718
BLEU-4	**88.618**	88.477
ROUGE-L Precision	**0.969**	0.968
ROUGE-L Recall	**0.969**	0.967
ROUGE-L F1	**0.969**	0.967

Table 2: Backbone comparison on the BanglaGov test set. Both use the same SpaMo-BdSL pipeline; only the language model differs. Higher is better for all metrics.

Table 2 shows the results. The two models are essentially tied: mT0-XL is marginally ahead on BLEU-3, BLEU-4, and ROUGE-L (for example, 88.618 vs. 88.477 on BLEU-4, about 0.14 points), while BanglaT5 leads on BLEU-1 and BLEU-2. A Bangla-specific model with roughly $15\times$ fewer parameters thus matches the large multilingual model on this dataset. The decisive difference is cost: BanglaT5 uses far less memory and translates the whole test set in 40 seconds versus 2 minutes 43 seconds for mT0-XL, about $4\times$ faster on the same hardware.

We read this as focus beating scale. A model trained only on Bangla has seen the script, spelling, and word patterns far more deeply than a multilingual model that spreads its capacity across more than a hundred languages. In a low-resource, narrow-domain setting, this focus lets a compact model keep pace with one an order of magnitude

larger. This echoes the finding that bigger models mostly help when there is enough high-quality data to feed them (Kaplan et al., 2020), and the original SpaMo result that scaling up the LLM did not always improve translation (Hwang et al., 2025). For an underrepresented language, then, a small language-specific model can match a much larger one while being far *more feasible* to train and deploy.

5 Discussion: Reflections for Underrepresented Sign Languages

We highlight three reflections that we hope can start a conversation about SLT for low-resource and underrepresented languages.

"Bigger and more multilingual" is not automatically more inclusive. Large multilingual models are often presented as the inclusive choice because they cover many languages at once. Our result complicates this story. For Bangla, a small dedicated model matched the quality of a much larger multilingual model while being far cheaper and faster to run. Coverage on paper is not the same as good service in practice. For an underrepresented sign language, investing in focused, community-relevant models may serve the language at least as well as adding it as one more entry in a giant multilingual model, and at a cost that local researchers can actually afford.

Compute access is an accessibility issue. If reproducing a result needs a data center, then researchers in low-resource regions, often the same regions whose sign languages are underrepresented, are pushed out. The changes in Section 3 show that meaningful work is still possible on modest hardware. We think reporting and sharing such "low-compute recipes" should be a normal part of low-resource SLT research, not an afterthought.

Narrow datasets give high scores but narrow promises. Because no prior baseline exists for BanglaGov, we contextualize our numbers with PHOENIX14T (Camgoz et al., 2018), a German weather-forecast benchmark. Running our SpaMo-BdSL pipeline with Flan-T5-XL (Chung et al., 2024) on PHOENIX14T reproduces the original SpaMo numbers (Hwang et al., 2025) closely (BLEU-4 19.53 versus 24.32), which indicates that our compute adaptations (Section 3) did not break the system. As Table 3 shows, our BanglaGov BLEU-4 (88.48) is roughly four times the PHOENIX14T BLEU-4. We do not read this as being four times "better." It mainly reflects how narrow and repetitive BanglaGov is: a small vocabulary and highly similar eating-related sentences make the task far easier than open-domain weather reports. Our scores should therefore not be read as general BdSL translation. Inclusive evaluation needs diverse signers, dialects, and everyday topics, gathered with the community's consent and involvement.

Metric	PHOENIX14T (Flan-T5-XL, original)	PHOENIX14T (Flan-T5-XL, ours)	BanglaGov (BanglaT5, ours)
BLEU-1	49.80	44.44	96.30
BLEU-2	37.32	31.56	93.79
BLEU-3	29.50	24.12	91.72
BLEU-4	24.32	19.53	88.48
ROUGE-L F1	46.57	39.47	96.70

Table 3: Our BanglaGov result (BanglaT5) next to PHOENIX14T results with Flan-T5-XL: both the numbers reported in the original SpaMo work (Hwang et al., 2025) and our own reproduction using our setup. The datasets and languages differ, so this is for context, not a fair head-to-head comparison. ROUGE-L F1 is reported on a 0–100 scale. The large gap mainly reflects how much narrower BanglaGov is.

6 Conclusion

We shared an early-stage adaptation of the gloss-free SpaMo framework for BdSL, built under real resource limits. Two findings stood out: a small Bangla-specific model matched a much larger multilingual model on our data while being far cheaper and faster to run, and a few simple, AI-suggested configuration changes made training possible on a single consumer-grade GPU. We offer these as talking points rather than final answers. For SLT to reach underrepresented languages and their communities, we believe the field should question the "bigger is better" default, treat compute access as part of accessibility, and evaluate systems on data that reflects the diversity of the communities they claim to serve.

Ethics Statement

This work aims to support communication access for the Deaf and hard-of-hearing community in Bangladesh. Automatic translations can be wrong, so such a system should support people rather than replace human interpreters, and should be used with a human in the loop for anything important.

Our model is trained on a narrow domain and a small set of signers, so it should not be assumed to work for all of BdSL or all signers. We encourage future data collection that includes more signers, dialects, and topics, with informed consent and community participation. We also note that we used an AI assistant (Claude Opus 4.6) to help choose training settings; we treat this as a tool whose suggestions we checked, not as an author.

References

BanglaGov. 2025. Bangla sign language sentence dataset. Available at https://huggingface.co/datasets/banglagov/Ban-Sign-Sent-9K-V1.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin Ahmad, and Rifat Shahriyar. 2023. BanglaNLG and BanglaT5: Benchmarks and resources for evaluating low-resource natural language generation in Bangla. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 726–735, Dubrovnik, Croatia. Association for Computational Linguistics.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7784–7793.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, and 16 others. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Necati Cihan Camgöz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10020–10030.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Eui Jun Hwang, Sukmin Cho, Junmyeong Lee, and Jong C. Park. 2025. An efficient gloss-free sign language translation using spatial configurations and motion dynamics with LLMs. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3901–3920, Albuquerque, New Mexico. Association for Computational Linguistics.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. 2022. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pretraining. In *Advances in Neural Information Processing Systems*, volume 35, pages 10078–10093. Curran Associates, Inc.

Benjia Zhou, Zhigang Chen, Albert Clapés, Jun Wan, Yanyan Liang, Sergio Escalera, Zhen Lei, and Du Zhang. 2023. Gloss-free sign language translation: Improving from visual-language pretraining. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20814–20824.

Zero-Shot Cross-Lingual Recognition of Sign Language Handshapes

Marcel Granero-Moya*, **Carolina del Corral Farrarós***,
Gloria Haro, **Coloma Ballester**, **Ricardo Marques**

Universitat Pompeu Fabra

Abstract

Sign language processing advances rapidly for high-resource languages such as American Sign Language (ASL), yet most of the world's sign languages lack the phonological annotations new methods require. We present the first zero-shot cross-lingual framework for handshape recognition, transferring from ASL to Catalan Sign Language (LSC). Our approach leverages the decomposition of handshapes into five phonological features — selected fingers, flexion, spread, thumb position, and thumb contact — shared across both languages, to decode LSC handshapes from predicted features via a composite phonological distance metric. We evaluate three architectures (MLP, SL-GCN, SHuBERT) trained on two ASL corpora (PopSign, Sem-Lex) against a 37-handshape, single-signer LSC benchmark. Zero-shot transfer proves viable once recording-format disparities are harmonized, reaching 80.0% phonological feature accuracy and 54.5% expected handshape accuracy. Phonological decomposition thus offers a bridge for extending sign language technologies to low-resource languages without any target-language video training labels.

1 Introduction

While over 70 million deaf people globally use over 200 distinct sign languages (SLs) (World Federation of the Deaf, 2026), most remain low-resource (Bragg et al., 2019). Annotated corpora and modeling efforts are heavily concentrated on a few high-resource languages like American Sign Language (ASL). In contrast, Catalan Sign Language (LSC) lacks the annotated resources needed to drive modern recognition systems, beyond a gloss-annotated dataset (Institut d'Estudis Catalans, 2025) and comprehensive grammar (Quer and Barberà, 2020).

Sign Language Processing (SLP) is bottlenecked by annotation costs: labels are costly and require fluent experts, and gloss-level supervision does not transfer across distinct lexicons. Phonological features offer a compelling solution. They provide a linguistically grounded label set that bridges languages. Because handshapes decompose into core phonological features—*selected fingers*, *finger flexion*, *spread*, *thumb position*, and *thumb contact* (Brentari, 1998)—a model trained on a high-resource language could theoretically recognize unseen target-language handshapes by recombining these shared phonological features.

While monolingual phonological recognition is well established (Kezar et al., 2023; Carbo and Nalisnick, 2025; Gueuwou et al., 2025; Inoue et al., 2026), and prior cross-lingual modeling relies on target-language annotations (Tornay et al., 2020; Bilge et al., 2024), strictly zero-shot cross-lingual phonological recognition remains unexplored.

We address this gap by transferring from ASL to LSC. Our contributions are as follows.

- The first zero-shot cross-lingual handshape recognition framework, evaluated across three architectural paradigms (static MLP, SL-GCN (Jiang et al., 2021), and SHuBERT (Gueuwou et al., 2025)), using two ASL corpora (PopSign (Starner et al., 2023) and Sem-Lex (Kezar et al., 2023)).
- A formal phonological alignment scheme mapping Navarrete-González's (2020) LSC handshapes into the ASL-LEX 2.0 (Sehyr et al., 2021) phonological feature space.
- A distance-based decoding protocol and expected-accuracy metric designed to handle mappings from phonological features to target-language handshapes.
- Empirical evidence that transfer is viable—achieving up to 80.0% phonological feature accuracy and 54.5% expected handshape accuracy—but only after harmonizing recording-format disparities.

* Equal contribution.
Correspondence: marcel.granero@upf.edu

Proceedings of the Second Workshop on Sign Language Processing, pages 177–192
October 29, 2026 ©2026 Association for Computational Linguistics

2 Related work

Our contribution bridges five lines of work that structure this section. Cross-lingual phonological transfer relies on three foundations: a shared feature inventory (*phonology of sign languages*), robust sign encoding models (*sign language processing*), and annotated resources (*phonology-curated datasets*). While their convergence enables monolingual *phonology recognition*, extending this to *cross-lingual phonological recognition* remains an open frontier—where we position this work.

Phonology of Sign Languages Stokoe (1960) and Battison (1978) established that Manual Features of signs decompose into contrastive sublexical units: handshape, palm orientation, location, and movement. Subsequent frameworks like Brentari's (1998) Prosodic Model refined these into a hierarchical feature geometry, decomposing handshapes into finer-grained features such as selected fingers, finger flexion, spread, and thumb position; and decomposing movement into path and location components. Modern datasets like ASL-LEX (Caselli et al., 2017; Sehyr et al., 2021) code phonology precisely at this granular level. Crucially, LSC relies on these same sub-handshape parameters (Quer and Barberà, 2020; Navarrete-González, 2020). This structural overlap, stemming from their shared lineage in the *Francosign* family (Wittmann, 1991), motivates the cross-linguistic transfer pursued in this work.

Sign Language Processing Early SLP relied on gloss-based pipelines constrained by comparatively small annotated datasets like RWTH-PHOENIX-Weather dataset (Forster et al., 2012, 2014; Camgoz et al., 2018, 2020). Recent advances have shifted the paradigm toward gloss-free translation (Zhou et al., 2023; Lin et al., 2023; Müller et al., 2023), mapping video or pose sequences directly to text, but critically depend on robust visual features. To address this, recent work leverages self-supervised pre-training on large, unannotated video corpora like YouTube-ASL (Uthus et al., 2023) and YouTube-SL-25 (Tanzer and Zhang, 2025). Models like SHuBERT (Gueuwou et al., 2025) use these datasets to learn generalized spatio-temporal representations without costly gloss annotations.

Phonology-Curated Datasets Interpreting fine-grained sub-lexical structures requires dedicated phonological resources. Several curated datasets have emerged, though available resources remain overwhelmingly restricted to ASL: ASL-LEX 2.0 (Sehyr et al., 2021) catalogs properties for over 2,700 glosses; Sem-Lex (Kezar et al., 2023) aligns deaf signers' isolated productions to ASL-LEX 2.0; and PopSign (Starner et al., 2023; Chow et al., 2023) offers game-collected sign recordings linked to these phonological features. We refer the reader to Section 3 for further dataset details.

Phonology Recognition Early work focused on isolated feature extraction, with milestones like DeepHand (Koller et al., 2016) targeting handshape recognition. To enable deeper phonological understanding, subsequent systems have leveraged on the aforementioned curated datasets to capture complex spatial and temporal variations. For example, Kezar et al. (2023) demonstrated that phonology recognition is an effective auxiliary target for Isolated Sign Language Recognition (ISLR), showing that an SL-GCN (Jiang et al., 2021) trained on Sem-Lex achieves 85% average accuracy across the 16 ASL-LEX phonological feature types. For handshape recognition, the Handshape-GNN model (Carbo and Nalisnick, 2025)—trained for the PopSign Kaggle challenge (Chow et al., 2023)—separates temporal sign dynamics from static hand configurations, benchmarking effectively against a multilayer perceptron (MLP) baseline. Similarly, the self-supervised SHuBERT (Gueuwou et al., 2025) achieved state-of-the-art ISLR accuracy on Sem-Lex when fine-tuned, suggesting that its learned representations implicitly encode sub-lexical phonological structure.

Cross-lingual Phonological Recognition Despite successes with rich, language-specific training data, the aforementioned models have not been evaluated on lower resourced SLs in a cross-lingual setup. Prior cross-lingual approaches adapt phonological subunits on target-language data (Tornay et al., 2020) or recognize novel signs from a few labeled target-language examples (Bilge et al., 2024). However, target-language annotation is a prohibitively expensive bottleneck for under-documented sign languages. A zero-shot approach bypasses this by leveraging shared cross-linguistic phonological structures. To our knowledge, zero-shot cross-lingual sign phonology prediction remains unexplored. We address this gap, demonstrating successful zero-shot transfer and establishing the first baseline for this task.

3 Datasets

To evaluate cross-lingual transfer across diverse capture conditions and signers, our framework combines three ASL resources with a target LSC benchmark. Table 1 synthesizes their primary characteristics.

ASL-LEX 2.0 (Sehyr et al., 2021) serves as the ASL lexical-phonological reference dictionary, providing over 2,700 glosses with detailed phonological annotations. While the dataset tracks 16 phonological features, our handshape prediction pipeline employs the 5 phonological features strictly related to the handshape (*selected fingers*, *flexion*, *spread*, *thumb position*, and *thumb contact*) plus handshape class. Crucially, the combination of these 5 features is not always a strict 1-to-1 mapping. As detailed by Sehyr et al. (2021), phonological feature combinations can group non-contrastive variants together or fail to differentiate contrastive handshapes that vary in unselected finger flexion (see Section 6).

Sem-Lex (Kezar et al., 2023) connects ASL-LEX 2.0 annotations to a large-scale corpus of over 84,000 webcam video instances representing 3,149 signs recorded remotely by 41 Deaf signers. The dataset combines ASL-LEX citation signs, free-text responses, and SignBank (Hochgesang et al., 2019) entries. Due to reproducibility issues with the official pre-extracted poses, we re-extracted all poses directly from the raw video source.

PopSign (Starner et al., 2023) introduces in-the-wild mobile recordings collected from 47 ASL learners, comprising ∼175k videos covering 250 isolated glosses balanced across all three dataset splits. We utilize the raw video release of the game split rather than the landmark-only Kaggle release (Chow et al., 2023), as RGB source videos are required for hybrid multimodal architectures (e.g., SHuBERT) that process pixel features alongside pose sequences.

LSC Benchmark[1] was accessed as a pre-release subset to evaluate zero-shot cross-lingual transfer. It comprises 37 target handshapes performed by a native Deaf LSC signer. Structured according to SignHub's LSC finger configuration (Navarrete-González, 2020), each handshape contains two videos: (1) a target handshape demonstration, and (2) an example sign.

[1]A pre-release subset was shared with us prior to publication; redistribution and visual sample display are restricted by the dataset creators.

Dataset	Lang.	#Hs	#Glosses	#Videos	#Signers	Rec. Type
ASL-LEX 2.0†	ASL	58	2,723	2,723	–	Studio
Sem-Lex	ASL	58	3,149	84,568	41 Deaf	Webcam
PopSign	ASL	37	250	175,022	47 Learners	Smartphone
LSC Benchmark	LSC	37	–	74	1 Deaf	Studio

Table 1: Overview and comparison of datasets evaluated in our experiments. Lang.: sign language; #Hs.: number of unique handshape classes; Rec. Type: video recording environment or device. †ASL-LEX 2.0 was used strictly as a lexical mapping reference; its video data was not utilized in our experiments.

4 Methodology

4.1 Model Architectures

To benchmark phonological feature prediction and cross-lingual transferability, we evaluate three representative architectural paradigms: a lightweight pose-based MLP baseline, a spatio-temporal graph neural network (SL-GCN) (Jiang et al., 2021), and a multi-modal pre-trained transformer (SHuBERT) (Gueuwou et al., 2025). Each architecture is trained and evaluated across two distinct ASL source datasets spanning different capture modalities: Sem-Lex and PopSign. To ensure complete reproducibility, we publicly release our model implementations, hyperparameter configurations, and evaluation pipelines, together with trained model checkpoints.[2]

4.1.1 Baseline MLP

Following the static baseline architecture introduced by Carbo and Nalisnick (2025), we employ a Multi-Layer Perceptron (MLP) with 3 layers of dimensions $63 \rightarrow 256 \rightarrow 256 \rightarrow N_{\text{classes}}$. The model operates strictly on static spatial features, removing the temporal dimension by extracting a single representative frame per sign sequence corresponding to minimal hand motion. Input features consist of raw 3D hand landmarks ($21 \times 3 = 63$ features) extracted via MediaPipe (Lugaresi et al., 2019). The network is trained from scratch on Sem-Lex and PopSign independently to predict multi-label phonological feature targets. This model serves as a computationally efficient reference to compare with more complex temporal and multi-modal architectures.

4.1.2 SL-GCN

To capture dynamic spatio-temporal hand movements and finger interactions, we employ the

[2]https://marcelgranero.github.io/cross-lingual-handshapes

Sign Language Graph Convolutional Network (SL-GCN) (Jiang et al., 2021), which served as the primary baseline for the Sem-Lex benchmark (Kezar et al., 2023). SL-GCN models pose sequences as spatio-temporal graphs, applying graph convolutions across skeletal joints and temporal convolutions across consecutive frames. Following Kezar et al.'s (2023), we first train the network on ISLR before fine-tuning for joint gloss and phonological feature prediction across both Sem-Lex and PopSign. To maintain a unified extraction pipeline across both datasets (Section 3) and achieve optimal pose fidelity, all input pose sequences are re-extracted via MediaPipe (Lugaresi et al., 2019) directly from the raw video sources. The complete reproduction benchmarks and validation metrics are provided in Table 10 (Appendix C).

4.1.3 SHuBERT

To test whether large-scale video-pose pre-training enhances zero-shot cross-lingual transfer compared to purely pose-based models, we evaluate SHuBERT (Gueuwou et al., 2025). SHuBERT is a self-supervised multimodal transformer pre-trained on YouTube-ASL (Uthus et al., 2023) that fuses raw video pixels with pose sequences. We fine-tune the pre-trained encoder for phonological feature prediction on Sem-Lex and PopSign. Reproduction performance benchmarks against the original paper are detailed in Table 11 (Appendix C).

4.2 Zero-Shot Cross-Lingual Transfer

To evaluate zero-shot handshape recognition across languages, we map target LSC handshapes to five phonological features in ASL-LEX format. Models trained on ASL predict this shared phonological feature space, and we map the resulting predicted features to target LSC handshape classes with a distance-based metric. Note that this protocol is zero-shot with respect to target-language videos and video-level labels, but it does presuppose a dictionary-level target resource: a predefined LSC handshape inventory and its handshape-to-feature mapping (Appendix A).

4.2.1 Aligning LSC Handshapes to ASL-LEX Phonological Features

To bridge the structural differences between ASL-LEX 2.0 and the SignHub LSC annotations (SH-LSC), we establish a rule-based alignment that deterministically assigns the five ASL-LEX phonological features pertaining strictly to handshape configuration —*selected fingers*, *finger flexion*, *spread*, *thumb position*, and *thumb contact*— to each LSC handshape (detailed in Appendices A and B). Specifically, Spread and Thumb Contact are mapped directly from SH-LSC to ASL-LEX categories without structural modifications. Finger flexion is mapped by harmonizing label conventions across schemes. For example, SH-LSC *Extended* is mapped to ASL-LEX *FullyOpen*. Additionally, because SH-LSC uses a single *curved* category, *curved* maps to a combined *Curved/Bent* class formed by joining ASL-LEX *Curved* and *Bent*. Regarding thumb phonology, SH-LSC treats the thumb as a standard selected finger, whereas ASL-LEX excludes the thumb from *Selected Fingers* unless it is the sole active digit. To align these paradigms, we map SH-LSC *Selected Fingers* onto ASL-LEX *Selected Fingers* and *Thumb Position*: if an SH-LSC handshape includes the thumb t[3] in its selected set, t is re-encoded as *Thumb Position* = *Open* and removed from *Selected Fingers*, unless t is the only selected digit (e.g., SH-LSC $\{t, i, m\} \rightarrow$ ASL-LEX $\{i, m\}$ + *Open*).

4.2.2 Phonological Handshape Matching and Evaluation

Phonological Representation Space Let $\mathcal{P}_{\text{sel}}, \mathcal{P}_{\text{flex}}, \mathcal{P}_{\text{sprd}}, \mathcal{P}_{\text{th\_pos}}, \mathcal{P}_{\text{th\_cnt}}$ be discrete categorical sets representing five articulatory features: *selected fingers*, *finger flexion*, *spread*, *thumb position*, and *thumb contact*, respectively. The phonological domain $\mathcal{P}$ is defined as their Cartesian product: $\mathcal{P} = \mathcal{P}_{\text{sel}} \times \mathcal{P}_{\text{flex}} \times \mathcal{P}_{\text{sprd}} \times \mathcal{P}_{\text{th\_pos}} \times \mathcal{P}_{\text{th\_cnt}}$ An arbitrary phonological configuration is thus represented as a 5-dimensional column vector $\mathbf{p} = [p_{\text{sel}}, p_{\text{flex}}, p_{\text{sprd}}, p_{\text{th\_pos}}, p_{\text{th\_cnt}}]^T \in \mathcal{P}$, where each component p_k takes a value from its corresponding feature set $\mathcal{P}_k$. Let $\mathcal{H} = \{h_1, h_2, \ldots, h_C\}$ denote the set of $C = 37$ target LSC handshape classes. Each class $h \in \mathcal{H}$ is associated with a canonical phonological feature vector $\mathbf{p}^{(h)} \in \mathcal{P}$ (See Table 7). We define the set of valid LSC representations as $\mathcal{P}_{\text{LSC}} = \{\mathbf{p}^{(h)} \mid h \in \mathcal{H}\} \subset \mathcal{P}$. Notably, the handshape-to-feature mapping $h \mapsto \mathbf{p}^{(h)}$ is non-injective: two pairs[4] among the 37 handshape

[3] Selected fingers are denoted as thumb t, index i, middle m, ring r, and pinkie p.

[4] `7-flat_open` with `index+thumb-flat_open` and `7-flat_closed` with `index+thumb-flat_closed` coincide on phonological features. These handshapes differ on the unselected fingers flexion – closed or open for `7` and `index+thumb` respectively.

classes share identical feature vectors, yielding $|\mathcal{P}_{\text{LSC}}| = 35$ unique canonical feature configurations. Combinations in $\mathcal{P}$ that do not correspond to any valid LSC handshape form the set of unassociated phonological vectors $\mathcal{U} = \mathcal{P} \setminus \mathcal{P}_{\text{LSC}}$.

Inference and Constrained Decoding During inference, the ASL-trained models generate categorical predictions for the five phonological channels. Because ASL's *selected fingers* and *finger flexion* features contain more classes than those in LSC (see Appendix B), we restrict each model's predictions to valid LSC target classes. We achieve this by applying a target-space mask to the output logits prior to the argmax operation[5], yielding a predicted feature vector $\hat{\mathbf{p}} = [\hat{p}_{\text{sel}}, \hat{p}_{\text{flex}}, \hat{p}_{\text{sprd}}, \hat{p}_{\text{th\_pos}}, \hat{p}_{\text{th\_cnt}}]^T \in \mathcal{P}$. However, since each feature is predicted by an independent classification head, nothing prevents the joint prediction $\hat{\mathbf{p}}$ from falling into $\mathcal{U}$.

Composite Phonological Distance Metric Given a discrete predicted feature vector $\hat{\mathbf{p}} \in \mathcal{P}$ and a target handshape vector $\mathbf{p}^{(h)} \in \mathcal{P}_{\text{LSC}}$, we measure their dissimilarity using a composite distance function $d(\hat{\mathbf{p}}, \mathbf{p}^{(h)}) = \sum_{c \in \mathcal{C}} d_c(\hat{p}_c, p_c^{(h)})$ where $\mathcal{C} = \{\text{sel}, \text{flex}, \text{sprd}, \text{th\_pos}, \text{th\_cnt}\}$ represents the set of feature categories. The category distances d_c are defined as follows. First, d_{sel} is the Hamming distance between the two selected-finger representations, where a mismatch (present vs. absent) for any of the four individual fingers (i, m, r, p) adds 1 to the distance. Second, d_{flex} is defined over the four flexion values (*FullyOpen*, *Flat*, *Curved/Bent*, and *FullyClosed*), assigning a distance of 1 to every distinct pair except the extreme transition FullyOpen $\leftrightarrow$ FullyClosed, which costs 2. Finally, d_{sprd}, $d_{\text{th\_pos}}$, and $d_{\text{th\_cnt}}$ assign a distance of 0 for matching values[6] and 1 for mismatches.

Candidate Selection and Evaluation The model outputs discrete categorical class predictions for each channel. Rather than requiring an exact vector match, we map the predicted feature vector $\hat{\mathbf{p}}$ to a set of candidate target handshapes $\hat{\mathcal{H}} \subset \mathcal{H}$ by retrieving all classes whose feature vectors minimize $d(\hat{\mathbf{p}}, \mathbf{p}^{(h)})$: $\hat{\mathcal{H}} = \text{argmin}_{h \in \mathcal{H}} d(\hat{\mathbf{p}}, \mathbf{p}^{(h)})$. Because multiple handshapes can tie for the minimum distance, $\hat{\mathcal{H}}$ is a candidate set containing one or more class labels. We evaluate performance using *Expected Accuracy*. Let $h^* \in \mathcal{H}$ denote the true target class. Sample-level expected accuracy $A(h^*, \hat{\mathcal{H}})$ measures the probability of selecting h^* under a uniform random choice from $\hat{\mathcal{H}}$, defined as $A(h^*, \hat{\mathcal{H}}) = 1/|\hat{\mathcal{H}}|$ if $h^* \in \hat{\mathcal{H}}$, and 0 otherwise. Table 2 illustrates a prediction $\hat{\mathbf{p}} \in \mathcal{U}$ mapped to a candidate set $\hat{\mathcal{H}}$ of three equidistant target classes ($d = 1$)[7]. This instance scores $1/3$ if $h^* \in \hat{\mathcal{H}}$, and 0 otherwise.

Candidates (h)	**Phonological Features** ($\mathbf{p}^{(h)}$)	$d(\hat{\mathbf{p}}, \mathbf{p}^{(h)})$
Prediction ($\hat{\mathbf{p}} \in \mathcal{U}$)	$\{i, m, r\}$, FullyOpen, 1, Open, 0	–
`8-extended`	$\{i, m\}$, FullyOpen, 1, Open, 0	1
`5-extended`	$\{i, m, r, p\}$, FullyOpen, 1, Open, 0	1
`3-extended`	$\{i, m, r\}$, FullyOpen, 1, Closed, 0	1
`2-extended`	$\{i, m\}$, FullyOpen, 1, Closed, 0	2
`4-extended`	$\{i, m, r, p\}$, FullyOpen, 1, Closed, 0	2
`5-curved-open`	$\{i, m, r, p\}$, Curved/Bent, 1, Open, 0	2
`8-curved-open`	$\{i, m\}$, Curved/Bent, 1, Open, 0	2
`B-extended`	$\{i, m, r, p\}$, FullyOpen, 0, Open, 0	2

Table 2: Candidate selection for a prediction in $\mathcal{U}$. Three target classes tie at minimum distance $d = 1$, forming $\hat{\mathcal{H}}$. Next-nearest classes ($d = 2$) are excluded. Mismatched features are in red.

4.3 Mitigating Recording-Format Disparities

The LSC benchmark consists of uniform 16:9 1080p full-body studio footage recorded at 50 fps. In contrast, the training corpora rely on in-the-wild recordings—webcam video for Sem-Lex and smartphone footage for PopSign—with both training sets having a median frame rate near 30 fps. To isolate cross-lingual errors from these recording-format disparities, we modify the LSC Benchmark clips to match the training capture geometries and subsample them to lower frame rates. To address spatial disparities, we extract a static bounding box centered on the median shoulder position for each target LSC clip, scaling it to match the training corpus medians. Specifically, the target normalized shoulder width and vertical y-position are aligned with Sem-Lex's 4:3 landscape webcam format (denoted as webcam) and PopSign's 3:4 portrait smartphone format (denoted as phone). Finally, to address temporal disparities, frame rate subsampling is applied to the LSC target clips after feature extraction using strides of 2 (25 fps) and 3 (16.7 fps). Integer strides avoid frame interpolation; 25 fps approaches the training corpora's frame-rate medians, while 16.7 fps probes robustness below them. Note that

[5] We mask three *flexion* classes (*Crossed*, *Stacked*, and *NA*) and four *selected fingers* combinations (*imp*, *mr*, *mrp*, and *r*).

[6] We include the NA class as a distinct, valid state for Spread and Thumb contact.

[7] See SignHub's Finger Configuration.

harmonization uses no target-side supervision: the crop is computed independently for each clip from its own pose sequence together with source-corpus constants, without labels or any fitting across the test set, and is therefore applicable to any input video at inference time.

5 Results

In this section, we empirically evaluate the cross-lingual transferability of phonological representations from ASL to LSC. We first verify our baseline source-task reproductions and address pose extraction artifacts. Next, we evaluate zero-shot transfer performance on the LSC benchmark for the three model architectures (MLP, SL-GCN, and SHuBERT) and the two training corpora (Sem-Lex and PopSign). Finally, we present a series of ablations to isolate the key drivers of transfer performance, examining the impact of recording-format alignment, auxiliary supervision heads, and SHuBERT's input channels and pretraining.

Source-Task Reproduction and Pose Correction Table 3 compares our reproduced performance on Sem-Lex against the metrics reported for SL-GCN (Kezar et al., 2023) and SHuBERT (Gueuwou et al., 2025). While our SHuBERT reproduction closely matches the published results, reaching 80.2% test macro Recall@1 versus the reported 79.4%, SL-GCN gloss recognition exhibits a noticeable gap. Training on the released Sem-Lex pose files yields 57.1% Top-1 accuracy, underperforming the reported 66.6% by 9.5 percentage points. However, when the model is trained with the poses we extracted from the source videos—instead of Sem-Lex released poses—and following the identical recipe, gloss top-1 accuracy increases to 72.2%, surpassing the published baseline rather than falling 9.5 points below it. Then, we fine-tune it for joint gloss and phonological feature prediction and improve the phonological feature average from 78.7% to 87.1%. The shortfall could be a property of the released data. Elsewhere in this paper the SL-GCN Sem-Lex rows refer to the model trained with re-extracted poses. Further reproduction details are provided in Appendix C.

Zero-shot transfer to LSC We train all three model architectures on PopSign and Sem-Lex using the full set of 16 phonological features. This choice maintains consistency with prior work and we hypothesize that additional feature targets provide auxiliary supervision without degrading performance, as ablated later in this section. However, since our cross-lingual handshape recognition approach relies only on five handshape-related phonological features, we evaluate performance using the mean accuracy across these five phonological features (Ph.% ↑) and the expected handshape accuracy (Hs.% ↑). All models are evaluated across three random seeds both in-domain on the corresponding ASL test splits and zero-shot on LSC, reporting mean performance alongside standard deviations (Table 4).

	Gloss Top-1 (%) ↑			16-Feat. Ph. (%) ↑		
Model	**Ref.**	**Ours**	**Δ**	**Ref.**	**Ours**	**Δ**
SL-GCN (released poses)	66.6	57.1	-9.5	85.0	78.7	-6.3
SL-GCN (re-posed)	66.6	72.2	+5.6	85.0	87.1	+2.1
SHuBERT	–	–	–	79.4	80.2	+0.9

Table 3: Reproduction fidelity of published Sem-Lex-trained source models on the test set. Gloss Top-1: top-1 sign gloss classification accuracy; 16-Feat. Ph.: mean accuracy across all 16 phonological features. Ref. denotes published baselines from Kezar et al. (2023) for SL-GCN and Gueuwou et al. (2025) for SHuBERT. The Δ columns give absolute percentage point differences between our reproduction and the reference.

As anticipated, all models score significantly lower in the zero-shot LSC setting than in-domain, with both metrics exhibiting similar trends. Training on Sem-Lex transfers better to unseen LSC handshapes than training on PopSign, particularly for the MLP (11.8% vs. 27.3% Hs.) and SHuBERT (13.9% vs. 18.3% Hs.) models. For SL-GCN, the two training corpora yield indistinguishable zero-shot performance (8.3% vs. 8.6% Hs.). Notably, while the MLP is the weakest model in-domain, it emerges as the *strongest* zero-shot model on LSC, leading both transfer metrics (62.6% Ph., 27.3% Hs.). We hypothesize that high-capacity spatio-temporal models overfit to source-dataset biases and pose extraction artifacts.

Mitigating the recording-format gap As outlined in Section 4.3, we re-evaluated each model under recording conditions closer to its training distribution to isolate cross-lingual transfer from recording-format disparities. To maintain a strict zero-shot source-only setup, harmonization is determined by the training corpus alone: each model is evaluated with spatial cropping matching its capture geometry and the frame rate closest to its training median—webcam at 25 fps for Sem-Lex-

Train	Model	ASL test		LSC zero-shot	
		Ph. (%) ↑	Hs. (%) ↑	Ph. (%) ↑	Hs. (%) ↑
PopSign	MLP	76.5±0.1	55.4±0.1	53.6±1.6	11.8±1.7
	SL-GCN	**98.2±0.1**	**96.5±0.1**	52.3±0.7	8.3±2.4
	SHuBERT	94.3±0.1	89.7±0.3	**56.7±1.5**	**13.9±0.7**
Sem-Lex	MLP	75.2±0.2	48.8±0.2	**62.6±1.7**	**27.3±1.2**
	SL-GCN	**90.0±0.1**	**78.4±0.3**	47.9±2.6	8.6±3.7
	SHuBERT	81.7±0.2	64.9±0.2	59.9±1.3	18.3±4.2

Table 4: In-domain ASL vs. zero-shot LSC transfer performance. Three model families are trained on PopSign and Sem-Lex, then evaluated on their respective ASL test splits and zero-shot on the LSC benchmark. Ph.: mean accuracy across handshape phonological features; Hs.: expected handshape accuracy. Cells report the mean ± standard deviation across random seeds. **Bold** indicates the best-performing model within each training corpus block.

trained models and phone at 25 fps for PopSign-trained ones (shaded cells in Table 5). Under this corpus-matched setting, the Sem-Lex MLP reaches 80.0% Ph. / 54.5% Hs., the Sem-Lex SHuBERT 65.6% / 22.1%, and the PopSign SL-GCN 70.4% / 29.5%—gains of +17.4 / +27.2, +5.7 / +3.8, and +18.1 / +21.2 points (Ph. / Hs.), respectively, over their uncropped 50 fps baselines (Table 4).

For completeness and as an ablation of both factors, Table 5 also reports performance across the full grid of crop and frame-rate combinations. Spatial cropping drives the vast majority of the gains: compared to uncropped baselines at matched frame rates, webcam and phone cropping yield average improvements of +11.1 / +11.3 and +10.5 / +10.6 (Ph. / Hs.), respectively, whereas frame-rate reduction offers marginal benefits (≤ 0.7 points averaged across models; see Table 13 in Appendix D).

Crucially, tuning the configuration on the target benchmark—rather than adhering to a strict zero-shot protocol—yields minimal gain. The grid optimum coincides with the corpus-matched setting for the strongest model (Sem-Lex MLP). For the remaining models, the grid optima exceed the corpus-matched configuration by at most 2.4 Ph. / 2.7 Hs. points (e.g., Sem-Lex SHuBERT at webcam, 50 fps: 66.1% / 23.5%; PopSign SL-GCN at phone, 16.7 fps: 72.5% / 31.5%). Thus, the observed improvements hold under a strictly zero-shot, source-only setup.

Appendix E decomposes these results per phonological feature: the corpus-matched harmonization gains concentrate on selected fingers and spread, while flexion remains the weakest feature.

	Model	Crop	Ph. (%) ↑			Hs. (%) ↑		
			50	25	16.7	50	25	16.7
PopSign	MLP	no crop	53.6	50.4	51.9	11.8	10.4	12.0
		webcam	71.9	**72.6**	71.9	25.3	26.9	**27.8**
		phone	70.7	70.2	69.5	26.0	26.4	25.9
	SL-GCN	no crop	52.3	55.0	54.7	8.3	11.8	9.9
		webcam	67.0	68.2	69.3	19.8	22.4	20.7
		phone	70.4	70.4	**72.5**	26.0	29.5	**31.5**
	SHuBERT	no crop	56.7	55.6	54.8	13.9	13.4	14.5
		webcam	55.6	56.2	56.8	14.6	15.8	15.1
		phone	55.1	**56.9**	56.4	14.9	**15.8**	13.5
Sem-Lex	MLP	no crop	62.6	60.5	62.9	27.3	22.4	25.0
		webcam	79.7	**80.0**	79.8	53.8	**54.5**	52.9
		phone	76.5	76.9	76.3	42.7	43.7	42.8
	SL-GCN	no crop	47.9	50.2	51.6	8.6	9.6	10.3
		webcam	56.3	57.2	57.8	13.1	14.5	15.9
		phone	54.3	58.2	**59.1**	13.1	**17.2**	15.2
	SHuBERT	no crop	59.9	58.5	58.2	18.3	15.7	16.3
		webcam	**66.1**	65.6	65.0	**23.5**	22.1	23.5
		phone	64.6	63.9	63.6	22.7	20.2	22.5

Table 5: Zero-shot evaluation on the LSC benchmark of video-crop (none, 4:3 webcam, 3:4 phone) and frame-rate (50, 25, 16.7) ablations for models trained on PopSign and Sem-Lex. Ph.: mean accuracy of five handshape phonological features; Hs.: expected handshape accuracy. **Bold** indicates the best Ph. and Hs. result per model configuration. Shaded cells mark the corpus-matched harmonization configuration—crop matching the training corpus, 25 fps (nearest to the training median)—fixed without consulting LSC results.

Target phonological heads To ablate the effect of auxiliary supervision, we retrained each model family on both ASL corpora to predict only the five handshape phonological features alongside the handshape class, rather than all 16 phonological heads. Holding model architectures and random seeds fixed, we present these comparative results in Table 15 (Appendix F). Removing auxiliary supervision yields no systematic benefits. In-domain, the shallow MLP is largely unaffected (≤ 0.3 percentage point shift), while deeper architectures suffer minor-to-moderate declines in handshape accuracy, particularly on Sem-Lex (SHuBERT −4.8, SL-GCN −3.1). On zero-shot LSC transfer, performance fluctuates directionlessly (e.g., SHuBERT Ph. +2.9 vs. SL-GCN Ph. −4.7 on Sem-Lex) with overlapping seed standard deviations in nearly all cases. We conclude that while auxiliary heads offer mild regularization for deep models in-domain, they neither systematically aid nor degrade cross-lingual transfer.

	Ablation	ASL test		LSC zero-shot	
		Ph. (%) ↑	Hs. (%) ↑	Ph. (%) ↑	Hs. (%) ↑
PopSign	all channels (reference)	**94.3±0.1**	**89.7±0.3**	**56.7±1.5**	**13.9±0.7**
	hands-only	**94.3±0.1**	89.4±0.3	56.2±2.0	**13.9±3.8**
	body-only	55.4±0.0	12.3±0.0	50.3±0.0††	2.7±0.0††
	no pre-training	90.6±0.5	83.1±0.9	50.6±4.6	8.6±3.5
Sem-Lex	all channels (reference)	81.7±0.2	**64.9±0.2**	59.9±1.3	18.3±4.2
	hands-only	**81.8±0.1**	64.2±0.6	**64.4±3.2**	**23.5±5.8**
	body-only	55.0±0.0	11.0±1.1	50.3±0.0††	2.7±0.0††
	no pre-training	74.3±0.3	50.7±1.0	58.4±1.4	14.9±1.5

Table 6: Input channel and pre-training ablation for SHuBERT fine-tuned on PopSign and Sem-Lex, evaluated in-domain (ASL) and zero-shot (LSC). Ph.: mean phonological feature accuracy; Hs.: expected handshape accuracy. Cells report mean ± std over seeds; **bold** denotes best result per training corpus. ††Degenerate runs emitting constant majority-class predictions.

SHuBERT ablations We ablate SHuBERT along two axes (input channels and pretraining), varying one factor at a time against the all-channel pretrained reference (Table 6). **Input channels:** SHuBERT fuses four positional streams: face, left hand, right hand, and body posture. We evaluate configurations retaining all four channels, the two hand streams only, and the body stream only. Dropped channels are masked using SHuBERT's pretrained mask embedding. Restricting the input to hand channels matches the all-channel reference with performance gaps in-domain falling within the standard deviation, demonstrating that face and body pose contribute minimally to handshape recognition. Conversely, the body-only model collapses: without hand signals, it predicts the training majority class, yielding a constant-predictor floor (marked †† in Table 6). **Pretraining:** training SHuBERT from scratch without self-supervised pretraining confirms that SSL pretraining provides substantial performance gains across all setups. On Sem-Lex, pretraining boosts in-domain phonological feature macro accuracy by $+7.4$ points and handshape accuracy by $+14.2$ points, whereas on PopSign the corresponding gains are $+3.7$ and $+6.6$ points. Consequently, zero-shot transfer to LSC exhibits a corresponding degradation when removing pre-training.

6 Discussion

Cross-Lingual Transfer and Phonological Universality Our findings show that zero-shot cross-lingual transfer from ASL to LSC is viable once domain gaps in recording format such as spatial cropping are explicitly harmonized. This supports the hypothesis that phonological features capture foundational articulatory properties that transcend individual sign languages. However, important scope limitations remain. Our evaluation is restricted to a single historically related language pair (ASL $\rightarrow$ LSC) and a benchmark containing two videos per handshape from a single signer across 37 handshape classes (out of more than 43 described in LSC, excluding fingerspelling, numerals, and classifiers). In particular, signer-independent generalization within LSC cannot be established from this benchmark, and robustness to uncontrolled target recording conditions remains untested. Although our decoding framework remains mathematically agnostic to the target set $\mathcal{P}_{\text{LSC}}$, future work will evaluate this approach across larger multi-signer datasets and typologically unrelated sign language families to test cross-lingual universality.

Model Capacity, Dataset Biases, and Practical Efficiency A key insight from our experiments is the counterintuitive strength of the shallow MLP, which outperforms deeper architectures in zero-shot transfer. We hypothesize that high-capacity spatio-temporal models (such as SHuBERT and SL-GCN) overfit to source-dataset biases and pose-extraction artifacts. Because the MLP lacks temporal mechanics and receives only dominant-hand features, its capacity bottleneck prevents it from learning non-transferable contextual cues, yielding superior zero-shot generalization. Crucially, our ablations show that SHuBERT can be stripped to hand-only input streams and supervised on only six handshape-relevant heads without sacrificing accuracy. This stream pruning significantly reduces the annotation burden and computational overhead required to deploy phonological recognizers for handshape prediction in new target languages.

Static Phonological Representation Although the 5-dimensional phonological feature vector successfully grounds categorical decoding, its formulation reveals structural limits as observed by Sehyr et al. (2021). First, the feature mapping is non-injective: the five phonological categories used in this work fail to capture unselected finger flexion, causing distinct handshapes to map to identical phonological feature vectors. Second, reducing signs to a single static handshape simplifies dynamic signing reality, overlooking handshape transitions over time as well as non-dominant hand configurations. Extending the shared feature space to parameterize unselected fingers, dynamic tempo-

ral trajectories, and two-handed signs to the feature space would expand cross-lingual transfer across the complete manual phonological spectrum.

Expected Accuracy and Decoding Ties These representational constraints are directly manifested during evaluation, defining the expected handshape accuracy as a diagnostic metric rather than a standard classification performance score. Because candidate ties are resolved uniformly at random, even an ideal feature predictor achieves a theoretical ceiling of 94.6% (reflecting 35 unique canonical feature configurations across 37 target classes). Such ties stem from two distinct mechanisms: *structural ties* between handshape class pairs sharing identical phonological feature vectors, and *distance ties*, where a prediction lies equidistant from multiple valid LSC handshape classes. Future work can resolve distance ties by leveraging the model's per-feature confidences—ensuring a single, deterministic prediction—and disambiguate structural ties only through supplementary information, such as unselected finger flexion features.

7 Conclusions

In this work, we introduced the first zero-shot cross-lingual framework for sign language handshape recognition, bridging American Sign Language (ASL) and Catalan Sign Language (LSC) through a shared, linguistically grounded phonological feature space. By decomposing handshapes into five phonological features and mapping predictions via a composite distance metric, our method enables direct target-language classification without requiring target-language phonological training labels.

Evaluated across three distinct model architectures (MLP, SL-GCN, and SHuBERT) and two source corpora (PopSign and Sem-Lex), our experiments on a single-signer LSC benchmark show that zero-shot transfer is viable—achieving up to 80.0% phonological feature accuracy and 54.5% expected handshape accuracy. Crucially, we show that isolating true cross-lingual transfer requires first harmonizing recording-format disparities such as spatial cropping. Our experiments reveal that low-capacity models (MLP) offer superior zero-shot generalization, providing evidence that model complexity does not necessarily translate into better cross-lingual transfer; we hypothesize that high-capacity models overfit source-dataset spatio-temporal biases. Complex multimodal transformers (SHuBERT), meanwhile, can be pruned to hand-only streams without loss of handshape performance. These results support the hypothesis that phonological decomposition can provide a language-agnostic representation bridge: for the language pair studied, this work substantially reduces target-language annotation requirements and opens new avenues for scaling sign language technologies to underserved sign communities and low-resource sign languages worldwide.

Acknowledgments

This work was supported by the Maria de Maeztu Units of Excellence Programme (CEX2021-001195-M, funded by MICIU/AEI/10.13039/501100011033), the PULSAR project (Ref. PID2025-173459NB-C22, financed by MICIU/AEI/10.13039/501100011033 and FEDER/UE), the MuReLA project (Ref. PID2025-173709NB-I00, funded by MICIU/AEI/10.13039/501100011033), and VISTA-SL (2024-1-EL01-KA220-HED-000257847) funded by EU's Erasmus+ programme. M. G. was supported by the AGAUR-FI Joan Oró predoctoral grant (2024 FI-3 00065) from the Secretariat for Universities and Research of the Department of Research and Universities of the Generalitat de Catalunya and the European Social Fund Plus. G. H. acknowledges support from the Serra Húnter Programme (Generalitat de Catalunya) as a Serra Húnter Associate Professor. We acknowledge the EuroHPC Joint Undertaking for awarding us access to MareNostrum5 at BSC (Spain) and Leonardo at CINECA (Italy).

We would like to thank Josep Blat and Laia Tarrés for their advice and support, Lee Kezar for early discussions on the topic, UPF's LSC-Lab and Lali Ribera for their willingness to collaborate, and in particular Alexandra Navarrete for guiding our understanding of the phonology of Catalan Sign Language.

References

Robbin Battison. 1978. *Lexical Borrowing in American Sign Language*. Linstok Press, Silver Spring, MD.

Yunus Can Bilge, Nazli Ikizler-Cinbis, and Ramazan Gokberk Cinbis. 2024. Cross-lingual few-shot sign language recognition. *Pattern Recognition*, 151:110374.

Danielle Bragg, Oscar Koller, Mary Bellard, Larwan Berke, Patrick Boudreault, Annelies Braffort, Naomi Caselli, Matt Huenerfauth, Hernisa Kacorri, Tessa

Verhoef, et al. 2019. Sign language recognition, generation, and translation: An interdisciplinary perspective. In *Proceedings of the 21st international ACM SIGACCESS conference on computers and accessibility*, pages 16–31.

Diane Brentari. 1998. *A Prosodic Model of Sign Language Phonology*. MIT Press, Cambridge, MA.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7784–7793. Ieee.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10023–10033.

Alessa Carbo and Eric Nalisnick. 2025. Improving handshape representations for sign language processing: A graph neural network approach. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 29122–29135, Suzhou, China. Association for Computational Linguistics.

Naomi K Caselli, Zed Sevcikova Sehyr, Ariel M Cohen-Goldberg, and Karen Emmorey. 2017. ASL-LEX: A lexical database of american sign language. *Behavior research methods*, 49(2):784–801.

Ashley Chow, Glenn Cameron, Mark Sherwood, Phil Culliton, Sam Sepah, Sohier Dane, and Thad Starner. 2023. Google - Isolated Sign Language Recognition.

Jens Forster, Christoph Schmidt, Thomas Hoyoux, Oscar Koller, Uwe Zelle, Justus H Piater, and Hermann Ney. 2012. Rwth-phoenix-weather: A large vocabulary sign language recognition and translation corpus. In *LREC*, volume 9, pages 3785–3789.

Jens Forster, Christoph Schmidt, Oscar Koller, Martin Bellgardt, and Hermann Ney. 2014. Extensions of the sign language recognition and translation corpus rwth-phoenix-weather. In *LREC*, pages 1911–1916.

Shester Gueuwou, Xiaodan Du, Greg Shakhnarovich, Karen Livescu, and Alexander H. Liu. 2025. SHuBERT: Self-supervised sign language representation learning via multi-stream cluster prediction. *Preprint*, arXiv:2411.16765.

Julie A. Hochgesang, Onno Crasborn, and Diane Lillo-Martin. 2019. ASL Signbank. New Haven, CT: Haskins Lab, Yale University.

Jundai Inoue, Daisuke Hara, and Makoto Miwa. 2026. A resource and evaluation method for phonological continuity in Japanese Sign Language. In *Proceedings of the Fifteenth Language Resources and Evaluation Conference*, pages 9514–9524, Palma de Mallorca, Spain. ELRA Language Resource Association.

Institut d'Estudis Catalans. 2025. Corpus de referència de la llengua de signes catalana (LSC) (CORPUS LSC). `https://corpuslsc.iec.cat/`.

Songyao Jiang, Bin Sun, Lichen Wang, Yue Bai, Kunpeng Li, and Yun Fu. 2021. Skeleton aware multi-modal sign language recognition. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3408–3418. IEEE.

Lee Kezar, Jesse Thomason, Naomi Caselli, Zed Sehyr, and Elana Pontecorvo. 2023. The sem-lex benchmark: Modeling asl signs and their phonemes. In *Proceedings of the 25th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 1–10.

Oscar Koller, Hermann Ney, and Richard Bowden. 2016. Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3793–3802.

Kezhou Lin, Xiaohan Wang, Linchao Zhu, Ke Sun, Bang Zhang, and Yi Yang. 2023. Gloss-free end-to-end sign language translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12904–12916.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. 2019. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*.

Mathias Müller, Zifan Jiang, Amit Moryossef, Annette Rios Gonzales, and Sarah Ebling. 2023. Considerations for meaningful sign language machine translation based on glosses. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 682–693.

Alexandra Navarrete-González. 2020. Phonology: 1. sublexical structure. In Josep Quer and Gemma Barberà, editors, *A Grammar of Catalan Sign Language (LSC)*, 1st edition, SIGN-HUB Sign Language Grammar Series.

Josep Quer and Gemma Barberà, editors. 2020. *A Grammar of Catalan Sign Language (LSC)*, 1st edition. SIGN-HUB Sign Language Grammar Series.

Zed Sevcikova Sehyr, Naomi Caselli, Ariel M Cohen-Goldberg, and Karen Emmorey. 2021. The ASL-LEX 2.0 project: A database of lexical and phonological properties for 2,723 signs in american sign language. *The Journal of Deaf Studies and Deaf Education*, 26(2):263–277.

Thad Starner, Sean Forbes, Matthew So, David Martin, Rohit Sridhar, Gururaj Deshpande, Sam Sepah, Sahir Shahryar, Khushi Bhardwaj, Tyler Kwok, et al. 2023.

Popsign asl v1. 0: An isolated american sign language dataset collected via smartphones. *Advances in Neural Information Processing Systems*, 36:184–196.

William C. Stokoe. 1960. Sign language structure: An outline of the visual communication systems of the American deaf. *Studies in Linguistics, Occasional Papers*, 8.

Garrett Tanzer and Biao Zhang. 2025. Youtube-sl-25: A large-scale, open-domain multilingual sign language parallel corpus. In *International Conference on Learning Representations*, volume 2025, pages 81921–81934.

Sandrine Tornay, Marzieh Razavi, and Mathew Magimai Doss. 2020. Towards multilingual sign language recognition. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6309–6313. IEEE.

Dave Uthus, Garrett Tanzer, and Manfred Georg. 2023. Youtube-asl: A large-scale, open-domain american sign language-english parallel corpus. *Advances in Neural Information Processing Systems*, 36:29029–29047.

Henri Wittmann. 1991. Classification linguistique des langues signées non vocalement. *Revue québécoise de linguistique théorique et appliquée*, 10(1):215–288.

World Federation of the Deaf. 2026. Faq: Deaf communities and sign languages. Accessed: 2026-08-21.

Benjia Zhou, Zhigang Chen, Albert Clapés, Jun Wan, Yanyan Liang, Sergio Escalera, Zhen Lei, and Du Zhang. 2023. Gloss-free sign language translation: Improving from visual-language pretraining. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20814–20824. IEEE.

A LSC Handshape Inventory and Feature Mapping

LSC handshapes were derived from the SignHub LSC Grammar: Finger Configuration. We inferred selected fingers from the finger configuration table, relying either on explicit mentions in the handshape names or on standard number and letter conventions. Configuration values correspond to the original header row, while spread was manually annotated following ASL-LEX rules.

Table 7 presents the complete inventory of 43 LSC handshapes alongside their original SignHub annotations and mapped ASL-LEX features. The deterministic alignment procedure used to link these two feature spaces is detailed in Section 4.2.1. Note that spread distinguishes class *5* (spread) from *B* (unspread) exclusively for *extended* and *curved open* configurations. The remaining configurations collapse into a single *5B* class. We encode this merged class as unspread based on SignHub's LSC finger configuration illustrations and the benchmark clips.

As noted by Sehyr et al. (2021) and discussed in Section 6, the five ASL-LEX handshape features—*selected fingers*, *flexion*, *spread*, *thumb position*, and *thumb contact*—cannot distinguish all handshapes, as certain classes differ exclusively in the flexion of unselected fingers.

	SignHub-inferred LSC annotations			ASL-LEX 2.0 phonological features				
LSC Handshape	**Sel. fingers**	**Configuration**	**Spread**	**Sel. fingers**	**Flexion**	**Spread**	**Thumb pos.**	**Thumb cont.**
6-extended	t	extended	NA	t	FullyOpen	NA	Open	NA
6-curved_open	t	curved open	NA	t	Curved/Bent	NA	Open	NA
6-closed	t	closed	NA	t	FullyClosed	NA	Open	NA
1-extended	i	extended	NA	i	FullyOpen	NA	Closed	0
1-curved_open	i	curved open	NA	i	Curved/Bent	NA	Closed	0
1-closed	i	closed	NA	i	FullyClosed	NA	Closed	1
middle-extended	m	extended	NA	m	FullyOpen	NA	Closed	0
middle-flat_open	m	flat open	NA	m	Flat	NA	Closed	0
i-extended	p	extended	NA	p	FullyOpen	NA	Closed	0
i-curved_open*	p	curved open	NA	p	Curved/Bent	NA	Closed	0
7-extended	ti	extended	NA	i	FullyOpen	NA	Open	0
7-flat_open†	ti	flat open	NA	i	Flat	NA	Open	0
7-flat_closed‡	ti	flat closed	NA	i	Flat	NA	Open	1
7-curved_open	ti	curved open	NA	i	Curved/Bent	NA	Open	0
index+thumb-flat_open†	ti	flat open	NA	i	Flat	NA	Open	0
index+thumb-flat_closed‡	ti	flat closed	NA	i	Flat	NA	Open	1
index+thumb-curved_closed	ti	curved closed	NA	i	Curved/Bent	NA	Open	1
middle+thumb-flat_open*	tm	flat open	NA	m	Flat	NA	Open	0
middle+thumb-flat_closed*	tm	flat closed	NA	m	Flat	NA	Open	1
middle+thumb-curved_closed*	tm	curved closed	NA	m	Curved/Bent	NA	Open	1
2-extended	im	extended	spread	im	FullyOpen	1	Closed	0
2-curved_open	im	curved open	spread	im	Curved/Bent	1	Closed	0
2-curved_closed	im	curved closed	spread	im	Curved/Bent	1	Closed	1
n-extended	im	extended	unspread	im	FullyOpen	0	Closed	0
n-curved_open*	im	curved open	unspread	im	Curved/Bent	0	Closed	0
Y-extended	tp	extended	NA	p	FullyOpen	NA	Open	0
u-extended	ip	extended	NA	ip	FullyOpen	NA	Closed	0
8-extended	tim	extended	spread	im	FullyOpen	1	Open	0
8-flat_closed	tim	flat closed	spread	im	Flat	0	Open	1
8-curved_open	tim	curved open	spread	im	Curved/Bent	1	Open	0
4-extended	imrp	extended	spread	imrp	FullyOpen	1	Closed	0
4-flat_open	imrp	flat open	spread	imrp	Flat	0	Closed	0
5-extended	timrp	extended	spread	imrp	FullyOpen	1	Open	0
5-curved_open	timrp	curved open	spread	imrp	Curved/Bent	1	Open	0
b-extended	timrp	extended	unspread	imrp	FullyOpen	0	Open	0
b-curved_open	timrp	curved open	unspread	imrp	Curved/Bent	0	Open	0
5B-flat_open	timrp	flat open	unspread**	imrp	Flat	0	Open	0
5B-flat_closed	timrp	flat closed	unspread**	imrp	Flat	0	Open	1
5B-curved_closed	timrp	curved closed	unspread**	imrp	Curved/Bent	0	Open	1
5B-closed	timrp	closed	unspread**	imrp	FullyClosed	NA	Open	1
pinkie+middle+thumb-extended	tmp	extended	NA	mp	FullyOpen	NA	Open	0
3-extended	imr	extended	spread	imr	FullyOpen	1	Closed	0
3-curved_open*	imr	curved open	spread	imr	Curved/Bent	1	Closed	0

Table 7: Mapping of the 43 LSC handshape classes (Navarrete-González, 2020) to ASL-LEX 2.0 phonological features, alongside their SignHub-inferred annotations. Asterisks (*) mark the six handshape classes lacking visual clips in the benchmark. The double asterisk (**) denotes collapsed *5B* classes encoded as unspread (see text). Classes sharing a dagger (†, ‡) possess identical ASL-LEX feature configurations, differing only in the flexion of unselected fingers.

B Phonological Feature Inventories Across Datasets

To analyze the structural granularity of the feature spaces across datasets, Table 8 contrasts the class counts across each phonological feature for Sem-Lex, PopSign, and the LSC Benchmark. Table 9 provides an itemized comparison of the value inventories for the five primary handshape features, explicitly denoting merged classes (e.g., *Curved/Bent*) and structural gaps between ASL-LEX and LSC.

Comparing these label spaces highlights a key structural divergence in *Selected Fingers*: the combination *mp* (middle and pinkie) is attested in LSC (e.g., in the *pinkie+middle+thumb-extended* handshape) but absent from the ASL-LEX training set. Consequently, our model cannot directly predict the exact *mp* value during zero-shot evaluation. Nevertheless, because target handshape retrieval relies on a distance metric over the combined feature space, *pinkie+middle+thumb-extended* can still be successfully recovered within the candidate set. To systematically handle unobserved finger combinations in future work, finger selection could be decomposed into independent per-digit binary features rather than treating joint configurations as discrete atomic classes.

Phonological Feature	**Sem-Lex**	**PopSign**	**LSC Benchmark**
Other phonological features			
Major Location	5	5	–
Minor Location	37	25	–
Second Minor Location	37	16	–
Contact	2	2	–
Sign Type	6	5	–
Repeated Movement	2	2	–
Path Movement	8	7	–
Wrist Twist	2	2	–
Spread Change	3	3	–
Nondominant Handshape	56	25	–
Handshape phonological features			
Selected Fingers	12	9	9
Flexion	8	7	4
Spread	3	3	3
Thumb Position	2	2	2
Thumb Contact	3	3	3
Handshape class			
Handshape	58	37	37

Table 8: Number of classes per phonological feature in each dataset's label space. Sem-Lex uses the full ASL-LEX 2.0 coding; PopSign's space is reduced to the values attested among its 250 glosses; the LSC Benchmark column is the canonical target space of the five handshape features in ASL-LEX format (Curved and Bent flexion merged) plus the 37 handshape classes – the remaining features have no LSC annotation (–). Counts include the NA class where the annotation has structurally-absent values (Table 9 lists the value inventories of the five handshape features).

Feature	**ASL-LEX 2.0 values**	**LSC values (ASL-LEX format)**
Selected Fingers	t, i, m, **r**, p, im, ip, **mr**, imr, **imp**, **mrp**, imrp	t, i, m, p, im, ip, **mp**, imr, imrp
Flexion	FullyOpen, Flat, **Curved**, **Bent**, FullyClosed, **Stacked**, **Crossed**, **NA**	FullyOpen, Flat, **Curved/Bent**, FullyClosed
Spread	0, 1, NA	0, 1, NA
Thumb Position	Open, Closed	Open, Closed
Thumb Contact	0, 1, NA	0, 1, NA

Table 9: Value inventories of the five phonological features strictly related to handshape: the ASL-LEX 2.0 space the models are trained on vs the reduced LSC target space in ASL-LEX format. **Bold** marks the changes: ASL-LEX values with no identical LSC counterpart (Crossed, Stacked, and flexion NA have no LSC realisation; Curved and Bent collapse into the merged Curved/Bent class; four selected-finger combinations are unattested in LSC) and LSC values absent from the ASL-LEX space (the merged Curved/Bent; mp, attested in LSC but not in ASL-LEX). NA is a real class, marking structurally-absent values, never missing supervision.

	Test Top-1 Accuracy (%) ↑				
		Released Poses		Re-posed	
Feature	**Ref.**	**Ours**	**Δ**	**Ours**	**Δ**
Gloss ISR (Tab. 4/6): the feature heads' encoder					
Gloss top-1	66.6	57.1	-9.5	72.2	+5.6
Gloss top-3	81.5	71.1	-10.4	85.6	+4.1
Phonological features (Tab. 5, multitask)					
Major Location	87.5	79.4	-8.1	88.4	+0.9
Minor Location	78.1	67.3	-10.8	80.0	+1.9
2nd Minor Location	77.2	69.2	-8.0	80.4	+3.2
Contact	88.6	85.1	-3.5	90.7	+2.1
Sign Type	87.9	80.8	-7.1	88.8	+0.9
Repeated Movement	85.4	81.5	-3.9	88.6	+3.2
Path Movement	75.4	69.7	-5.7	80.4	+5.0
Wrist Twist	92.6	89.5	-3.1	93.5	+0.9
Spread Change	89.5	84.3	-5.2	91.2	+1.7
Nondominant Handshape	81.7	72.4	-9.3	82.9	+1.2
Selected Fingers	90.2	84.6	-5.6	92.3	+2.1
Flexion	81.0	73.8	-7.2	83.3	+2.3
Spread	88.0	81.7	-6.3	89.6	+1.6
Thumb Position	91.5	87.7	-3.8	92.6	+1.1
Thumb Contact	91.1	86.4	-4.7	92.2	+1.1
Handshape	74.7	65.6	-9.1	78.4	+3.7
Average	**85.0**	**78.7**	**-6.3**	**87.1**	**+2.1**

Table 10: SL-GCN reproduction fidelity on the test split. We compare the results reported by Sem-Lex (Kezar et al., 2023) against our reproduction runs. Our models are evaluated on two pose sets: the original *Released Poses* provided by the benchmark, and *Re-extracted*, which denotes the poses we independently extracted from the same videos. The Δ columns indicate the absolute percentage point difference between our reproduction and the baseline.

C Reproduction Results

To ensure the reliability of our baseline implementations, we reproduce experiments on both SL-GCN (Kezar et al., 2023) and SHuBERT (Gueuwou et al., 2025). Table 10 reports the reproduction fidelity for SL-GCN across two distinct pose inputs: the official pose files shipped with the Sem-Lex release versus pose sequences extracted directly from the raw dataset videos using our pipeline. Evaluating on the official released poses reveals a systematic performance drop (averaging -6.3 percentage points), whereas our re-extracted pose sequences reliably match and slightly exceed original paper metrics ($+2.1$ percentage points on average across phonological tasks).

Table 11 evaluates our fine-tuned SHuBERT baseline against the published Sem-Lex test results. Our reproduced SHuBERT model consistently matches or marginally outperforms published per-feature Recall@1 scores in all 16 phonological dimensions, achieving an average performance gain of $+0.86$ percentage points.

	Test Recall@1 (%) ↑		
Phonological Feature	**Ref.**	**Ours**	**Δ**
Major Location	84.77	85.03	+0.26
Minor Location	71.30	71.73	+0.43
2nd Minor Location	73.28	73.64	+0.36
Contact	86.84	87.35	+0.51
Sign Type	84.64	84.66	+0.02
Repeated Movement	82.65	84.01	+1.36
Path Movement	72.75	72.92	+0.17
Wrist Twist	90.58	91.48	+0.90
Spread Change	81.60	82.85	+1.25
Nondominant Handshape	76.32	76.72	+0.40
Selected Fingers	79.53	81.04	+1.51
Flexion	72.64	74.38	+1.74
Spread	79.42	80.38	+0.96
Thumb Position	86.04	87.03	+0.99
Thumb Contact	84.74	85.76	+1.02
Handshape	62.93	64.89	+1.96
Average	**79.38**	**80.24**	**+0.86**

Table 11: SHuBERT reproduction fidelity evaluated on the Sem-Lex test set. We compare the per-feature Recall@1 results reported in the original SHuBERT baseline (Gueuwou et al., 2025) against our fine-tuned reproduction. The Δ column indicates the absolute percentage point difference between our reproduction and the baseline.

D Further results for mitigation of recording-format gap

To provide a full accounting of stability and individual factor contributions in our domain-adaptation experiments, this section presents detailed breakdowns of the cropping format and frame-rate ablations. Table 12 extends Table 5 from the main text by reporting sample standard deviations across three random seeds for every cell in the 3×3 grid of crop configurations (uncropped, 4:3 webcam, and 3:4 phone portrait) and frame rates (50 fps, 25 fps, and 16.7 fps). Table 13 isolates the marginal treatment effects (Δ) of spatial cropping relative to uncropped inputs, as well as temporal frame-rate reductions relative to native 50 fps video.

Spatial alignment drives gains. Spatial cropping yields substantial performance improvements for all models ($+10.5\%$–$+11.3\%$ average gain), peaking at $+28.9\%$ for MLP handshape accuracy on Sem-Lex. This shows pose-based models are sensitive to video framing, though SHuBERT is less affected by uncropped inputs. Conversely, frame-rate reductions (from 50 fps to 25 or 16.7 fps) have a negligible impact ($\leq +0.7\%$ average change). This confirms that spatial domain shift—rather than temporal resolution—is the primary bottleneck in zero-shot cross-dataset transfer.

	Model	Crop	Ph. (%) ↑ 50 fps	25 fps	16.7 fps	Hs. (%) ↑ 50 fps	25 fps	16.7 fps
PopSign	MLP	no crop	53.6±1.6	50.4±2.6	51.9±2.1	11.8±1.7	10.4±2.8	12.0±2.9
		webcam crop	71.9±2.1	**72.6±2.0**	71.9±2.1	25.3±2.0	26.9±2.8	**27.8±3.7**
		phone crop	70.7±2.3	70.2±1.1	69.5±1.3	26.0±1.9	26.4±1.8	25.9±1.4
	SL-GCN	no crop	52.3±0.7	55.0±1.0	54.7±2.3	8.3±2.4	11.8±0.5	9.9±1.7
		webcam crop	67.0±1.4	68.2±1.8	69.3±0.7	19.8±2.2	22.4±3.3	20.7±2.2
		phone crop	70.4±1.1	70.4±0.9	**72.5±0.4**	26.0±1.6	29.5±2.7	**31.5±1.0**
	SHuBERT	no crop	56.7±1.5	55.6±1.0	54.8±2.3	13.9±0.7	13.4±1.1	14.5±1.6
		webcam crop	55.6±1.5	56.2±1.5	56.8±1.1	14.6±1.4	15.8±1.0	15.1±0.7
		phone crop	55.1±1.4	**56.9±1.3**	56.4±1.2	14.9±0.2	**15.8±1.6**	13.5±1.8
Sem-Lex	MLP	no crop	62.6±1.7	60.5±1.9	62.9±1.6	27.3±1.2	22.4±0.8	25.0±1.2
		webcam crop	79.7±1.2	**80.0±0.9**	79.8±0.7	53.8±2.6	**54.5±3.6**	52.9±3.3
		phone crop	76.5±1.2	76.9±1.1	76.3±0.4	42.7±1.9	43.7±2.2	42.8±2.3
	SL-GCN	no crop	47.9±2.6	50.2±3.1	51.6±4.4	8.6±3.7	9.6±2.1	10.3±4.0
		webcam crop	56.3±4.3	57.2±3.3	57.8±0.9	13.1±4.8	14.5±1.2	15.9±0.5
		phone crop	54.3±5.3	58.2±5.3	**59.1±1.0**	13.1±3.6	**17.2±4.4**	15.2±1.7
	SHuBERT	no crop	59.9±1.3	58.5±0.7	58.2±1.8	18.3±4.2	15.7±3.2	16.3±4.6
		webcam crop	**66.1±2.3**	65.6±2.0	65.0±1.7	**23.5±5.3**	22.1±1.8	23.5±5.1
		phone crop	64.6±0.7	63.9±1.3	63.6±1.9	22.7±2.5	20.2±1.1	22.5±3.6

Table 12: Video-crop × frame-rate ablation on the LSC benchmark with seed spreads: the appendix sibling of Table 5. Models are trained on PopSign and Sem-Lex and evaluated zero-shot on the LSC Benchmark. Ph. is the mean accuracy over the five handshape phonological features; Hs. is the expected handshape accuracy; decoding is constrained to the LSC inventory. The *Crop* column gives the type of image cropping applied to the LSC clips (none, webcam-style 4:3, or phone-style 3:4 portrait), with each model's input re-extracted from the cropped video; the numeric sub-columns give the video frame rate in fps (native 50, subsampled 25 and 16.7). **Bold** marks, for each trained model, its best Ph. and best Hs. cell over the whole grid; shaded cells mark the corpus-matched harmonization configuration (crop matching the training corpus, 25 fps), fixed without consulting LSC results. Cells are mean ± sample sd over three seeds.

	Model	Phonological feature macro Δ (%) ↑ webcam	phone	25	16.7	Expected handshape Δ (%) ↑ webcam	phone	25	16.7
PopSign	MLP	**+20.2±2.3**	+18.2±2.0	-1.0±0.4	-1.0±0.3	**+15.3±3.6**	+14.7±3.1	+0.2±0.3	+0.9±0.9
	SL-GCN	+14.2±1.9	**+17.1±1.2**	+1.3±1.3	+2.3±1.0	+11.0±2.8	**+19.0±2.5**	+3.2±2.9	+2.7±1.9
	SHuBERT	**+0.5±0.8**	**+0.5±0.5**	**+0.5±0.3**	+0.2±0.4	**+1.2±1.7**	+0.8±1.0	+0.5±0.4	-0.1±1.1
	mean	+11.6±10.1	**+11.9±9.9**	+0.2±1.2	+0.5±1.7	+9.1±7.2	**+11.5±9.5**	+1.3±1.7	+1.2±1.4
Sem-Lex	MLP	**+17.8±1.2**	+14.6±1.1	-0.5±0.4	+0.1±0.4	**+28.9±2.4**	+18.2±1.4	-1.0±0.5	-1.0±0.7
	SL-GCN	+7.2±3.4	**+7.3±4.7**	+2.3±1.9	+3.3±2.1	+5.0±3.1	**+5.6±4.1**	+2.2±1.0	+2.2±1.6
	SHuBERT	**+6.7±2.2**	+5.2±2.3	-0.9±1.1	-1.3±1.0	**+6.3±0.8**	+5.1±2.3	-2.2±1.7	-0.7±2.3
	mean	**+10.6±6.3**	+9.0±4.9	+0.3±1.8	+0.7±2.4	**+13.4±13.4**	+9.6±7.4	-0.3±2.3	+0.1±1.8
Mean	MLP	**+19.0±1.7**	+16.4±2.5	-0.7±0.4	-0.5±0.7	**+22.1±9.6**	+16.4±2.5	-0.4±0.9	-0.1±1.3
	SL-GCN	+10.7±4.9	**+12.2±6.9**	+1.8±0.7	+2.8±0.7	+8.0±4.2	**+12.3±9.4**	+2.7±0.7	+2.4±0.3
	SHuBERT	**+3.6±4.4**	+2.8±3.3	-0.2±1.0	-0.5±1.0	**+3.8±3.6**	+2.9±3.0	-0.8±1.9	-0.4±0.5
	mean	**+11.1±7.5**	+10.5±7.2	+0.3±1.3	+0.6±1.8	**+11.3±9.9**	+10.6±7.7	+0.5±2.0	+0.7±1.5

Table 13: Treatment effects of the crop × frame-rate grid, one row per trained model. *webcam* and *phone* are the crop's effect against the uncropped clips, averaged over the three frame rates; *25* and *16.7* are the frame rate's effect against native 50 fps, averaged over the three crops. *Mean* rows average the unrounded deltas of the rows in their scope (a corpus, a model family across both corpora, or all six models); their ± is the sd across the averaged rows, not a seed sd – every other ± is a sample sd over three seeds.

	Model	Config	Per-feature accuracy (%) ↑ SF	Flex.	Sprd.	Th.Pos.	Th.Cnt.	Ph.
Majority class			27.0	40.5	54.1	62.2	67.6	50.3
PopSign	MLP	no crop, 50	47.7	44.6	64.9	**64.4**	46.4	53.6
		phone, 25	**76.6**	**60.8**	**87.8**	60.8	**64.9**	**70.2**
	SL-GCN	no crop, 50	49.5	40.1	53.6	64.9	53.2	52.3
		phone, 25	**74.3**	**53.6**	**77.5**	**66.7**	**79.7**	**70.4**
	SHuBERT	no crop, 50	39.6	**44.1**	**64.9**	**67.6**	**67.1**	56.7
		phone, 25	**44.1**	**44.1**	63.5	67.6	65.3	**56.9**
Sem-Lex	MLP	no crop, 50	64.0	48.6	63.5	74.8	62.2	62.6
		webcam, 25	**88.3**	**69.8**	**87.8**	**77.5**	**76.6**	**80.0**
	SL-GCN	no crop, 50	43.7	33.8	45.0	61.3	55.9	47.9
		webcam, 25	**57.7**	**41.9**	**58.6**	**63.1**	**64.9**	**57.2**
	SHuBERT	no crop, 50	49.1	47.7	64.9	69.4	68.5	59.9
		webcam, 25	**57.2**	**49.5**	**68.0**	**75.7**	**77.5**	**65.6**

Table 14: Per-feature error analysis on the LSC benchmark: accuracy of each of the five phonological features and their mean (Ph.), computed over the 74 benchmark clips and three seeds. Each model is evaluated zero-shot on the raw recordings (no crop, native 50 fps) and with corpus-matched harmonization (crop matching the training corpus, 25 fps; shaded in Table 5); the two settings differ only in label-free video pre-processing. Majority class: a constant predictor answering the benchmark's most frequent value per feature. Bold: best config per model and feature. SF: selected fingers.

E Per-Feature Error Analysis

Table 14 decomposes the zero-shot LSC results into the five phonological features, reporting the accuracy of each feature together with their mean (Ph.). Each model is evaluated zero-shot on the raw recordings (no crop, 50 fps) and with its corpus-matched harmonization; the rows thus decompose exactly the aggregate values of Table 5. The *majority class* row reports the accuracy of a constant predictor answering the benchmark's most frequent value per feature. Because our models are trained on ASL corpora whose class priors differ from the benchmark's and never observe its distribution, this row is a reference line rather than an expected floor. Three patterns emerge. First, flexion—a four-way distinction of finger curvature—is the weakest or tied-weakest feature in every corpus-matched configuration. Second, harmonization gains concentrate on selected fingers and spread (e.g., +24.3 points each for the Sem-Lex MLP), while thumb position is nearly unaffected. Third, measured against the majority references, the learned signal is largest for selected fingers (88.3% vs. a 27.0% reference for the Sem-Lex MLP) and smallest for the two thumb features, whose accuracies stay within a few points of their references.

	Model	Setting	ASL Test Ph. (%) ↑	ASL Test Hs. (%) ↑	LSC Zero-Shot Ph. (%) ↑	LSC Zero-Shot Hs. (%) ↑
PopSign	MLP	all16	76.5±0.1	55.4±0.1	53.6±1.6	11.8±1.7
		heads6	**76.7±0.1**	**55.5±0.2**	**55.1±0.7**	**12.8±0.5**
	SL-GCN	all16	**98.2±0.1**	**96.5±0.1**	52.3±0.7	**8.3±2.4**
		heads6	98.0±0.1	96.1±0.1	**53.9±1.1**	5.4±2.3
	SHuBERT	all16	**94.3±0.1**	**89.7±0.3**	**56.7±1.5**	13.9±0.7
		heads6	93.7±0.2	88.4±0.5	55.5±3.6	**16.5±2.9**
Sem-Lex	MLP	all16	75.2±0.2	**48.8±0.2**	**62.6±1.7**	**27.3±1.2**
		heads6	**75.5±0.1**	**48.8±0.2**	59.1±1.6	**27.3±1.2**
	SL-GCN	all16	**90.0±0.1**	**78.4±0.3**	**47.9±2.6**	**8.6±3.7**
		heads6	88.8±0.3	75.3±0.4	43.2±4.9	5.0±1.0
	SHuBERT	all16	**81.7±0.2**	**64.9±0.2**	59.9±1.3	18.3±4.2
		heads6	79.9±0.2	60.1±0.3	**62.8±3.0**	**18.5±3.3**

Table 15: Supervision head-set ablation across all three model families. Models are grouped by training set (PopSign and Sem-Lex blocks) and evaluated on two test sets: in-domain (ASL Test) and zero-shot (LSC Zero-Shot). Setting: full 16-head training (`all16`) vs. 6 handshape-relevant heads (`heads6`). Ph.: mean accuracy across the handshape phonological features; Hs.: expected handshape accuracy. Cells report mean ± standard deviation across seeds; **bold** denotes the higher value per paired ablation setting.

F Ablation on Target Phonological Heads

To evaluate the impact of auxiliary multi-task supervision, we compare models trained to predict all 16 phonological features (denoted as `all16`) against those trained exclusively on the 6 handshape-related heads, i.e. the five handshape phonological features alongside the handshape class, (denoted as `heads6`), as introduced in Section 5.

Table 15 details the empirical results, revealing a clear trade-off between in-domain performance and zero-shot transfer capability. For in-domain ASL evaluation, the full 16-head auxiliary supervision (`all16`) acts as a valuable multi-task regularizer for the more complex architectures (SL-GCN and SHuBERT), consistently yielding higher accuracy than `heads6`. However, in zero-shot LSC transfer, restricting supervision to `heads6` frequently results in competitive or even superior performance. For example, Sem-Lex-trained SHuBERT improves from 59.9% to 62.8% (Ph.) when discarding the auxiliary non-handshape heads. Similarly, the simpler MLP baseline slightly benefits from the focused `heads6` setting across both datasets. This suggests that while full multi-tasking helps robustly model in-domain ASL phonology, focusing solely on handshape variables can sometimes prevent over-specialization to ASL's specific feature distributions, aiding cross-lingual generalization.

HandReader: Advanced Techniques for Efficient Fingerspelling Recognition

Pavel Korotaev[1] pkorotaevdm@gmail.com
Petr Surovtsev[1] pysurovtsev@daimld.tech
Alexander Kapitanov[1] aakapitanov@daimld.tech
Karina Kvanchiani[1] karinakvanciani@gmail.com
Alexander Nagaev[1] aonagaev@daimld.tech

[1]Department of Artificial Intelligence and Machine Learning Development, DAIMLD

Abstract

Fingerspelling is a significant component of Sign Language (SL), allowing the interpretation of proper names, characterized by fast hand movements during signing. Although previous works on fingerspelling recognition have focused on processing the temporal dimension of videos, there remains room for improving the accuracy of these approaches. This paper introduces HandReader, a group of three architectures designed to address the fingerspelling recognition task. HandReader$_{RGB}$ employs the novel Temporal Shift-Adaptive Module (TSAM) to process RGB features from videos of varying lengths while preserving important sequential information. HandReader$_{KP}$ is built on the proposed Temporal Pose Encoder (TPE) operating on keypoints as tensors. Such keypoint composition in a batch allows the encoder to pass them through 2D and 3D convolution layers, utilizing temporal and spatial information and accumulating keypoint coordinates. We also introduce HandReader$_{RGB+KP}$ – architecture with a joint encoder to benefit from RGB and keypoint modalities. Each HandReader model possesses distinct advantages: on the ChicagoFSWild and ChicagoFSWild+ datasets, HandReader models achieve the best results among methods trained on a single training set, including a new state-of-the-art result on ChicagoFSWild+.

1 Introduction

Sign Language (SL) facilitates communication among more than 430 million deaf and hard-of-hearing individuals worldwide[1]. Due to the limited quantity of SL interpreters, an automatic SL Recognition (SLR) system can bridge the communication barrier between deaf individuals and people without SL knowledge (National Deaf Center, 2022; Nimdzi Insights, 2021; Globo Language Solutions, 2023). Such systems find application in schools, hospitals, and other social environments.

Fingerspelling is an integral component of any sign language, used for parts of speech, such as prepositions, conjunctions, and interjections, proper names, and instances without specific signs, such as addresses, abbreviations, and surnames. For example, to sign the metro station's name, the SL user should sign it letter by letter, using the corresponding signs for each letter from the SL alphabet. Therefore, fingerspelling is characterized by rapid and swift movements, complicating its recognition by blurry video frames. The motivation to tackle the fingerspelling recognition task is supported by interest from major companies, such as Google, which organized a Kaggle competition (Chow et al., 2023) to address it.

Many prior approaches to fingerspelling recognition are based on neural networks capable of processing the temporal component. To convert a video which consists of a frame sequence $I = \{I_1, I_2, ..., I_n\}$ to a letter sequence $W = \{W_1, W_2, ..., W_k\}$, encoder-decoder architectures are typically utilized (Fayyazsanavi et al., 2024; Kumwilaisak et al., 2022; Shi et al., 2018). However, fingerspelling recognition, similar to Automatic Speech Recognition (ASR) and Optical Character Recognition (OCR), is complicated by the lack of alignment between W and I, as multiple frames can interpret a single letter. The Connectionist Temporal Classification (CTC) loss (Graves, 2012) function is widely used in ASR and OCR to address this challenge by facilitating the learning of alignment between input frames and their corresponding output sequences, without the need for explicitly annotated segmentations. So, there is an option to simplify the task and enhance methods' performance by employing not only RGB frames but also such modalities as depth maps, keypoints, or combinations of video frames and keypoints (Kang et al., 2015; Shi et al., 2019; Fayyazsanavi

[1]https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss

Proceedings of the Second Workshop on Sign Language Processing, pages 193–202
October 29, 2026 ©2026 Association for Computational Linguistics

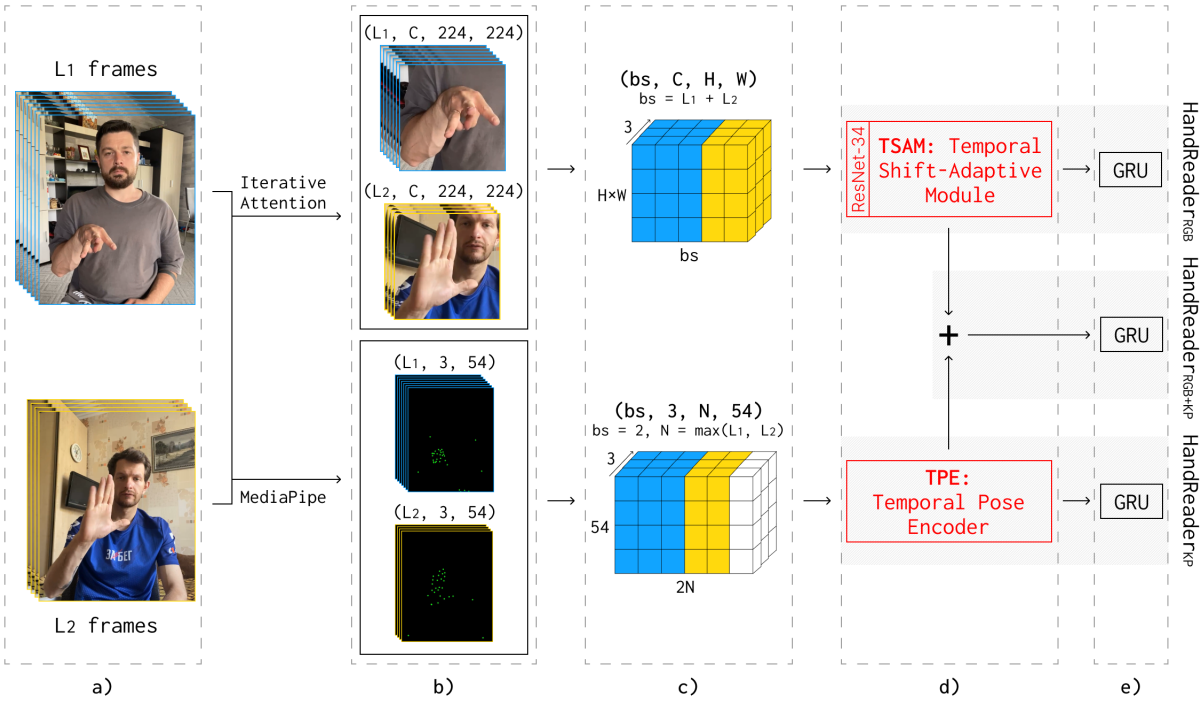


Figure 1: Three HandReader architectures. a) The videos with different lengths b) are processed to get crops and keypoints for HandReader$_{RGB}$ and HandReader$_{KP}$, respectively, c) form the batches, and d) pass through the encoders. Further, e) the GRU decoder is applied to the encoder outputs. The final output of the HandReader$_{RGB+KP}$ is obtained by summing the outputs of TSAM and TPE. Red-colored encoders are the paper's contribution.

et al., 2024; Parelli et al., 2020).

Despite multiple attempts to enhance fingerspelling recognition, the results of prior methods (Kang et al., 2015; Shi et al., 2019; Fayyazsanavi et al., 2024; Parelli et al., 2020; Shi et al., 2018; Pannattee et al., 2024; Kumwilaisak et al., 2022) on two American fingerspelling datasets – ChicagoFSWild (Shi et al., 2018) and ChicagoFSWild+ (Shi et al., 2019) highlight the need for improved accuracy to ensure the applicability of existing approaches in real-world scenarios. Our work aims to enhance letter recognition accuracy while keeping the architectures lightweight.

We aimed to develop an efficient approach for fingerspelling recognition. The contributions of this paper are summarized as follows:

- We introduce the Temporal Shift-Adaptive Module (TSAM), extending the original Temporal Shift Module (TSM) (Lin et al., 2019) to effectively process RGB image features from videos of varying lengths without trimming or padding.
- The Temporal Pose Encoder (TPE) to operate keypoints as three-channel tensors, capturing spatial and temporal information.
- HandReader, a group of three architectures (see Figure 1) that achieved a state-of-the-art result on the ChicagoFSWild+ test set and the best result on ChicagoFSWild among models trained solely on its training set.

We release the pre-trained models and code[2].

2 Related Work

This overview excludes isolated Sign Language Recognition and Continuous Sign Language Recognition (CSLR) research, as methods for these tasks operate at the sign level: CSLR is further complicated by the need to recognize the independent relationships between signs and letters within words. In contrast, fingerspelling recognition presents a more straightforward one-to-one correspondence.

Recently, the fingerspelling recognition task has been explored using various data modalities, including RGB imagery (Shi et al., 2019, 2018; Pannattee et al., 2024; Kumwilaisak et al., 2022), keypoints (Fayyazsanavi et al., 2024), and depth maps (Kang et al., 2015). Moreover, Parelli et al. (2020) utilized data obtained by combining different modalities. Researchers are motivated to focus on the

[2] https://github.com/ai-forever/HandReader

RGB modality because a fingerspelling recognition system must apply in real-world conditions and resist external changes. Meanwhile, acquiring data from other modalities requires additional equipment, such as extra cameras like Kinect for depth maps or an extra data processing stage for keypoints.

Typical videos for the fingerspelling task contain irrelevant information, such as backgrounds and foreign objects. Therefore, only RGB data requires the model to identify the most relevant areas in the frames, specifically the hands, before recognizing a sign sequence. To simplify locating the signing hand, Shi et al. (2018) developed a hand detection method using Faster R-CNN, then trained a CNN-LSTM model for the recognition task. In their subsequent work (Shi et al., 2019), they replaced the explicit hand detection method with an iterative attention mechanism. The authors of Pannattee et al. (2024) and Kumwilaisak et al. (2022) employed representation learning to improve the model's ability to distinguish different signs by alignment from CTC and incorporated the method from Shi et al. (2019) as a preprocessing stage. Pannattee et al. (2024) acquired state-of-the-art results on ChicagoFSWild and ChicagoFSWild+.

Since most RGB modality methods are centered around the hand detection problem, they are limited to exploring different architectures or decoding methods on the final results. In contrast, the keypoint modality avoids this issue because it includes relevant information about hand placement. Fayyazsanavi et al. (2024) trained a Transformer encoder-decoder architecture with a new decoding approach, advancing performance on ChicagoFSWild and ChicagoFSWild+ test sets. Keypoints are usually acquired using MediaPipe (Lugaresi et al., 2019) or another neural network trained for pose estimation. Parelli et al. (2020) combined RGB and keypoint modalities employing linear layers to project the 2D coordinates into 3D space. After extracting all features, they concatenated them and used a decoder based on an attention mechanism. Despite the promise shown by the methods above, recognition accuracy still has potential for improvement. Therefore, our primary objective is to enhance the methodologies employed without compromising the model's weight and inference speed.

Multi-Stream Sign Language Models. Combining RGB and keypoint streams has also been explored beyond fingerspelling. Zhou et al. (2021) exploited multiple visual cues, including hands, face, and full-frame appearance, for continuous sign language recognition and translation, while Chen et al. (2022) modeled raw videos and keypoint sequences with a two-stream network for the same tasks. More recently, self-supervised multi-stream approaches, such as SignMusketeers (Gueuwou et al., 2025a) and SHuBERT (Gueuwou et al., 2025b), learn sign language representations from separate face, hand, and body streams. These methods operate at the sign or sentence level and typically involve large-scale pre-training. In contrast, our work addresses the letter-level fingerspelling task, and its contribution lies at the encoder level: TSAM for variable-length RGB processing and TPE for tensor-based keypoint encoding, combined through a deliberately lightweight fusion.

RGB Feature Extraction in the Fingerspelling Task. There are two main ways to extract features in the fingerspelling task. (1) Spatial features can be received by applying a 2D CNN to each frame, then extracting temporal features using 1D convolutions or RNNs. (2) Spatio-temporal features can be obtained with 3D convolution. However, processing frames as a sequence of independent frames lacks temporal information. The TSM (Lin et al., 2019) was developed to address this issue by employing 2D convolutions and shifting information along the temporal dimension. However, TSM processes fixed-size videos, which leads to the loss of valuable information in long videos or negatively affects training time and memory allocation due to the padding in shorter videos. Therefore, this paper presents a modified version of the original TSM to handle videos of varying lengths.

3 Approach

We present a family of three architectures, collectively referred to as HandReader, designed for sign sequence recognition across multiple modalities. The models differ only in the encoder stage (see Figure 1). Thus, HandReader$_{RGB}$ solely utilizes RGB features, HandReader$_{KP}$ relies on keypoints, and HandReader$_{RGB+KP}$ combines both. Each was trained directly using CTC loss, eliminating the need for frame-level letter alignment. The CTC loss accounts for all possible alignments between hand shapes and sign language letters, eliminating the need for explicit alignment supervision. More formally:

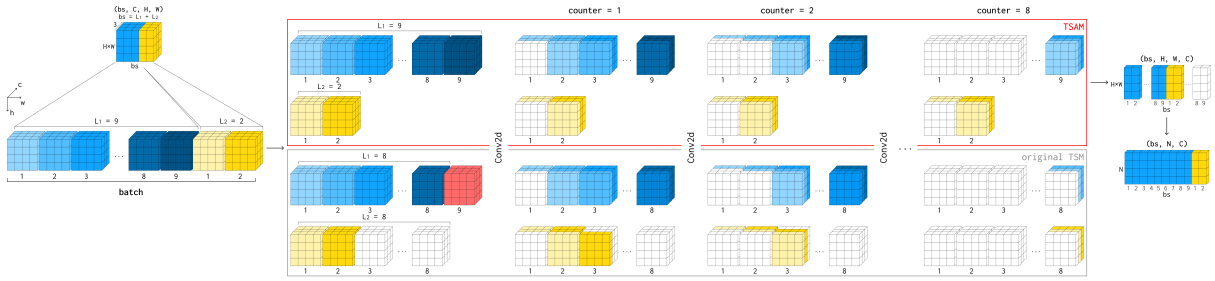


Figure 2: Comparison of the TSAM architecture (outlined in red at the top) with the original TSM (outlined in gray at the bottom). The input batch includes two videos of different lengths, with specific frames highlighted in blue and yellow. While the TSM model requires fixed-length inputs utilizing padding (shown in white) or trimming frames (shown in red) when necessary, the TSAM model employs individual shift counters to manage variable-length sequences without additional padding or trimming. The shift counter displayed at the top of the TSAM block shows how the shifts are activated individually for each video, incrementing the shift counter until it reaches the length of that specific video.

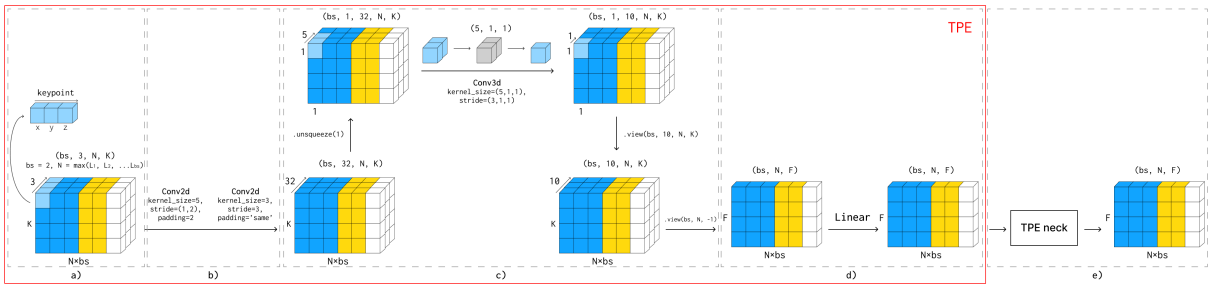


Figure 3: Demonstration of the Temporal Pose Encoder operating on a single batch. a) The encoder receives an input tensor containing N frames and K keypoints. All videos are padded to match the length of the most extended video clip in the batch, allowing them to be stacked. b) 2D convolutions process multiple keypoints across several frames, extracting temporal features. c) Subsequently, a 3D convolution in the form of a "tube" aggregates the extracted motion information for each body part into final feature representations F. d) After applying TPE, features are processed with linear layer, e) followed by MLP with convolution modules from Gulati et al. (2020).

$$L_{CTC} = -\log \sum_{A \in \mathcal{A}_{X,Y}} \prod_{t=1}^{T} p_t(a_t \mid X), \quad (1)$$

where $\mathcal{A}_{X,Y}$ represents the set of valid alignments between the model input X and the target sequence Y, $A = (a_1, \ldots, a_T)$ is one such alignment, and $p_t(a_t \mid X)$ denotes the probability of symbol a_t occurring at time step t. Each a_t is either a letter of the alphabet or the special blank symbol "_", introduced by CTC for time steps that emit no letter and to separate repeated letters; blanks and repetitions are removed from an alignment to obtain the final prediction.

A two-layer bidirectional Gated Recurrent Unit (GRU) (Cho et al., 2014) was chosen as a decoder to generate a sequence of characters, enhancing the model's capability to capture temporal dynamics. We apply a linear layer with SoftMax to the GRU output to generate posterior probabilities for each predicted character at every frame. Encoders for each architecture are described below.

3.1 HandReader$_{\text{RGB}}$

The original TSM is a video understanding approach that enables temporal modeling in 2D convolutional networks by shifting portions of channel features along the temporal dimension. However, TSM is constrained to processing a fixed number of frames. Therefore, videos that exceed the frame limit require trimming, while shorter videos need additional empty frames (i.e., padding). Furthermore, the number of shifts is determined by the fixed frame limit. As a result, videos with extra empty frames lose critical sequential information during processing (see Figure 2). We modified the TSM to manage a dynamic number of input frames, avoiding padding and enabling frame-by-frame processing for each video in the batch. Besides, a shift counter is introduced to preserve important sequential information, bypassing shifts on short videos.

Temporal Shift-Adaptive Module. The TSAM was designed to effectively process videos of various lengths. Figure 2 illustrates the distinction between TSAM and TSM operations. The TSAM is incorporated into the ResNet-34 model and operates an input tensor with dimension (bs, C, H, W), where bs indicates batch size, equal to the sum of the video lengths, C is the number of channels, H and W are the height and width of the frames, respectively. The TSAM also requires an auxiliary list of video lengths, denoted as $L = [L_{S_1}, L_{S_2}, \ldots, L_{S_n}]$, where S_i signifies ith video in the batch, and $bs = \sum_i L_{S_i}$. We iterate through L, extracting L_{S_i} frames from the batch to obtain S_i video sequence and apply shifts individually, as detailed in Algorithm 1. Such a method reduces VRAM usage by up to 50% during batch creation, thereby facilitating the processing of longer videos.

The number of shifts is determined by the number of convolutional blocks within the backbone architecture, leading to overwrite temporal features similar to the original TSM. To prevent this loss of information, we introduce a shift counter that stops temporal shifts at the final frame, i.e., no shifting is applied if a video's length is smaller than the current shift count. Such a method ensures that each video in a batch is treated individually, resulting in more accurate feature extraction.

Each sequence of the received tensor with dimension (bs, H, W, F), where F represents the number of features, was subsequently padded with zeros to align with the maximum sequence length within the batch. All features are aggregated using 1D convolution, and the resulting tensor is transformed into the format (bs, N, F), where F is the flattened features across all channels, and N is calculated as $H * W$.

Algorithm 1 Applying temporal shifts in TSAM.

Require: Input tensor S of size (bs, C, H, W), list of sequence lengths $L = [L_{S_1}, L_{S_2}, \ldots, L_{S_n}]$
Ensure: Output tensor with temporal shifts applied

```
for each sequence S_i in batch from input tensor do
    counter ← 0
    for each convolutional block in backbone do
        if counter < L_{S_i} then
            Apply temporal shift for sequence S_i
            counter ← counter + 1
        else
            Skip temporal shift for sequence S_i
        end if
    end for
    Apply 1D Convolution for S_i
end for
```

3.2 HandReader$_{\mathbf{KP}}$

We utilized MediaPipe (Lugaresi et al., 2019) to extract body keypoints from every frame. Face keypoints were excluded as stated in Section 5.2; we used 21 keypoints for each hand and 12 body pose keypoints (6 per arm), resulting in 54 (x, y, z) coordinates per frame. The proposed feature encoder is built on a new way to structure keypoints explicitly developed for it, which organizes keypoints within tensors. Each video is treated as a tensor with dimension $(3, N, 54)$, where 54 (x, y, z) vectors are stacked into matrices for each of N video frames. Such a configuration enables the spatial representation of temporal data, allowing the application of processing techniques typically used for standard images. We passed the received tensor to the feature extractor described below.

Temporal Pose Encoder. We present a Temporal Pose Encoder based on the convolution layers. TPE operates an input tensor of dimension $(bs, 3, N, K)$, where bs is the batch size, indicating the number of videos in the batch (in contrast to TSAM, where bs denotes the total number of frames across the videos in the batch), N denotes the maximum length of videos in the batch, while shorter videos were padded, and K – is the number of keypoint coordinates. The tensor is processed by two 2D convolution layers, followed by a 3D one (see Figure 3). 2D convolutions serve to extract both temporal and spatial features, while transforming keypoint coordinates, enhancing the model's understanding of dynamic finger movements. At the same time, 3D convolution tries to accumulate coordinate information within a single video with the kernel-tube size of $(5, 1, 1)$, treating each coordinate independently. In this context, the hyperparameter of 5 defines the size of the receptive window along the coordinate dimension, which captures local patterns across five neighboring coordinate features. We insert a dummy dimension in the second position, enabling 3D convolution to treat the original channels as temporal dimensions. The process can be repre-

Dataset	Lang	Sequences	Chars	Hours	Signers	Source
ChicagoFSWild	ASL	7K	38K	2	160	Web
ChicagoFSWild+	ASL	**55K**	0.3M	14	**260**	Web

Table 1: Open fingerspelling recognition datasets used in this work.

Augmentation	Parameters	Probability	Modality
Resample	rate=(0.5, 1.5)	0.8	KP, RGB+KP
SpatialAffine	scale=(0.8, 1.2), shear=(-0.15, 0.15), shift=(-0.1, 0.1), degree=(-30, 30)	0.75	KP
TemporalMask	size=(0.2, 0.4), value=0.0	0.5	KP
SpatialMask	size=(0.05, 0.1), value=0.0, mode="relative"	0.5	KP
HorizontalFlip	—	0.5	RGB, RGB+KP, KP
Rotation	angle=(-10, 10)	0.5	RGB, RGB+KP

Table 2: Augmentations used during training.

sented as follows: $(bs, 32, N, K)$ is unsqueezed to $(bs, 1, 32, N, K)$ and processed by 3D convolution to $(bs, 1, 10, N, K)$. A linear layer is then applied to project the features of the keypoints (bs, N, F), where F is the final encoder's features. We additionally use multi-layer perceptron (MLP) with convolution modules from Gulati et al. (2020) as a neck between the encoder and the GRU decoder.

3.3 HandReader$_{\text{RGB+KP}}$

We combine features from the HandReader$_{RGB}$ and HandReader$_{KP}$ encoders to take advantage of both modalities. Two output tensors of features from TSAM and TPE are summed and passed to the GRU (see Figure 1).

4 Experiments

This section outlines the experimental setup and data preprocessing procedures used to assess the proposed HandReader architectures and compare them with other approaches. Table 1 shows the open datasets used in this work.

4.1 Experimental Setup

All three architectures were trained from scratch using an AdamW optimizer (Loshchilov and Hutter, 2019) with an initial learning rate of 0.0001. The MultiStepLR modifies the learning rate with a gamma of 0.1 using milestones at 20 and 40 for the HandReader$_{RGB}$ and HandReader$_{RGB+KP}$ models and 25 and 40 for the HandReader$_{KP}$. We trained the HandReader$_{RGB}$ for 60 epochs, while the other two models were trained for 100 epochs each. A single Nvidia H100 with 80GB RAM was utilized for training.

4.2 Data Preprocessing

The ChicagoFSWild and ChicagoFSWild+ datasets are commonly used to evaluate approaches to the fingerspelling task.

The RGB frames for each dataset were cropped using the model described in Shi et al. (2019), augmented with horizontal flipping and random rotation (see parameters in Table 2), then resized to 224 and normalized. Keypoints were extracted by MediaPipe from raw frames rather than cropped images. We utilized only the hand keypoints (21 per hand) and 12 body pose keypoints (6 per arm), resulting in 54 keypoints. We replaced the NaN values with zeros, normalized the rest, and applied augmentations specified in Table 2. Only resampling, horizontal flipping, and random rotation augmentations are applied when combining two modalities in HandReader$_{RGB+KP}$.

4.3 Results

We assessed HandReader architectures using the letter accuracy metric: $1 - \frac{S+D+I}{N}$, where S, D, and I denote substitutions, deletions, and insertions, respectively, N is the length of ground-truth sequence. Metric values for other methods were obtained from the original papers for comparison purposes. Each HandReader version outperformed all prior methods trained on a single training set on both benchmarks, and HandReader$_{RGB+KP}$ achieved a new state-of-the-art result on ChicagoFSWild+ (see Table 3). Note that Conformers (CFW+CFW+) (Shi, 2023) reaches a higher score on ChicagoFSWild; however, it was trained on the union of the ChicagoFSWild and ChicagoFSWild+ training sets, whereas each HandReader model was trained only on the training set of the corresponding benchmark. We also present prediction comparison between PoseBased Transformer (Fayyazsanavi et al., 2024) and our models in Table 4. For decoding, all models employed the beam search strategy with a beam width of 5, which explores multiple candidate sequences at each step and selects the most probable transcription.

Table 5 provides qualitative examples of predictions produced by the three HandReader variants on the ChicagoFSWild dataset. We observe that the keypoint-based model (**KP**) tends to generate shorter and less stable outputs, frequently dropping characters (e.g., "youtube" → "outube") or producing heavily distorted strings. The RGB-based model (**RGB**) is generally more consistent

in capturing the global temporal structure, but still suffers from confusions between visually similar finger configurations, resulting in character-level substitutions (e.g., "organic" → "orgadic"). The multimodal model (**RGB+KP**) demonstrates the most robust behavior across the examples, often correcting errors present in the unimodal variants, which indicates that RGB and keypoint cues provide complementary information.

To further analyze the error patterns, Table 6 reports the SDI (Substitution, Deletion, Insertion) breakdown for each model under greedy and beam-search decoding. Across all models, **deletions** constitute the dominant error type, highlighting the difficulty of maintaining correct character continuity in fast fingerspelling sequences. Beam search consistently reduces the number of deletions for all variants (e.g., from 677 to 628 for RGB, and from 857 to 789 for KP), at the cost of a small increase in substitutions and insertions. This trade-off suggests that beam search improves sequence completeness but may introduce additional character-level confusions. Overall, the **RGB+KP** model achieves the best balance between the three error types, reinforcing the benefit of multimodal fusion for fingerspelling recognition.

5 Ablation

This section provides an ablation study to demonstrate the impact of each component on the HandReader's overall performance. Note that HandReader$_{RGB+KP}$ was excluded from the study since it relies solely on the heuristics of the other two architectures. We maintained a consistent experimental setup described in Section 4 and changed only one component at a time to analyze its influence. The ChicagoFSWild dataset was utilized in all experiments.

5.1 HandReader$_{\text{RGB}}$

Encoder. Table 7 indicates that integrating TSM into the baseline model improves the performance metric by 1.69. Furthermore, replacing TSM with TSAM leads to an additional 1.4 improvement. To isolate the effect of the count shift mechanism, we implemented a variant of TSAM without the timely stopping of shifts, which led to a 1.49 performance degradation.

Decoder. We experimented with the LSTM (Hochreiter and Schmidhuber, 1997) decoder instead of the GRU (Cho et al., 2014) one, which resulted in a difference of 1.69. Replacing RNN with a linear layer reduces the metric by 11.0. Such observations underlined that high-level features of each frame cannot be used as independent features for final predictions.

Others. Table 7 highlights the significant impact of augmentations, as a model trained without rotations and horizontal flips reached a final metric of 70.49. Such influence is explained by the dataset containing signers with different dominant hands and videos where hands are misaligned due to camera angles or the signer's position. Additionally, we tested different batch sizes and showed that a bigger batch size is not always optimal. It is especially noticeable in ablation studies with RGB. This may happen because the batch contains videos of varying lengths, complicating the model's decoding process.

5.2 HandReader$_{\text{KP}}$

Keypoints. We tested possible combinations of different parts of keypoints and provided results in Table 8. The metric was reduced by adding face keypoints since sign language users can express the same phrases differently without verbal language skills. The model performs better with pose and hand keypoints than with just hand keypoints, as the pose coordinates carry information about the position of the whole hand, which may help the model better understand the hand's relative position throughout the video.

Encoder and Decoder. The same procedure of RNN impact assessment is provided in Table 8. In contrast to Section 5.1, the usage of the linear layer does not show a significant metric drop, as the HandReader$_{KP}$ encoder contains more modules for processing temporal dimensions. We studied the impact of the number of convolution modules utilized in the encoder. As the number of convolution blocks increases, the final metric declines.

Others. The augmentations have a crucial effect on model performance. We also examined various batch sizes: 4, 6, and 8. As the batch size increases, there are noticeable metric drops.

5.3 HandReader$_{\text{RGB+KP}}$

Feature reduction. We evaluated various feature reduction techniques, including simple summation, feature concatenation, element-wise multiplication, and weighted summation, obtaining 72.95, 72.95, 72.55, and 72.57 letter accuracy, respectively. We preferred the simple summation due to the smaller

Model	ChicagoFSWild	ChicagoFSWild+	Modality
Hand Det + CNN + RNN (Shi et al., 2018)	41.9	41.2	RGB
Iterative Attention (Shi et al., 2019)	45.1	46.7	RGB
3d hand pose + RGB (Parelli et al., 2020)	47.9	–	RGB + KP
SiameseNet+Iterative model (Kumwilaisak et al., 2022)	49.6	–	RGB
Iter train + SupCon (Pannattee et al., 2024)	64.9	71.7*	RGB
Pose-based transformer (Fayyazsanavi et al., 2024)	66.3	71.1	KP
Conformers (CFW) (Shi, 2023)	68.4	60.2	RGB
Conformers (CFW+CFW+) (Shi, 2023)	77.5*	73.0*	RGB
HandReader$_{\mathbf{KP}}$	**69.3**	**72.4**	KP
HandReader$_{\mathbf{RGB}}$	**72.0**	**73.8**	RGB
HandReader$_{\mathbf{RGB+KP}}$	**72.9****	**75.6**	RGB + KP

Table 3: Results comparison on ChicagoFSWild and ChicagoFSWild+ test sets. * indicates the model was trained on both ChicagoFSWild and ChicagoFSWild+ training sets. ** means two convolution modules were used for HandReader$_{KP}$ encoder, while only one was employed for other metrics.

PoseBased (Fayyazsanavi et al., 2024)*	KP	RGB+KP	RGB	Reference
macah	mach	mach	mach	mach
meach	meah	beack	fealc	beach
hospital	hospital	hospital	hospital	hospital
-	outube	iutube	iutube	youtube
-	organic	organic	orgadic	organic

Table 4: Results comparison of predictions between previous SotA solution and our models, predictions for PoseBased model were taken from the original paper.

Gt	KP	RGB	RGB+KP
youtube	outube	iutube	iutube
issue	isue	isue	isue
organic	or ganic	orgadic	organic
pharmacology	opharmacoint	opharoacolohy	opharmacolgi
pharmacology	harmcogiy	haroscothy	harmacotgy
physical	phisical	phylsical	phistical
audio	uio	ardio	atudio
biochemistry	biochemitry	biockemistry	biochemiry

Table 5: Result of comparison of predictions between different models on ChicagoFSWild dataset.

dimension of the sum output than concatenation one. The evaluation was conducted on the test set of the ChicagoFSWild dataset.

6 Conclusion

This paper presents HandReader, a suite of three distinct architectures that significantly advance the fingerspelling recognition task. The Temporal Shift-Adaptive Module and Temporal Pose Encoder were developed to capture RGB and keypoint features effectively. We achieved state-of-the-art performance on the ChicagoFSWild+ dataset and the best performance on ChicagoFSWild among models trained solely on its training set. The ablation study revealed that the proposed techniques and training pipelines are essential for achieving optimal results. The presented findings offer valuable insights for enhancing fingerspelling recognition systems.

Model	D	S	I
RGB	677 / 628	462 / 478	122 / 130
KP	857 / 789	382 / 406	148 / 161
RGB+KP	683 / 621	386 / 409	160 / 164

Table 6: SDI breakdown on ChicagoFSWild. Each cell shows Greedy / Beam decoding.

Limitations

Architectures. Although the HandReader$_{KP}$ and HandReader$_{RGB}$ effectively recognize fingerspelling, extra models are required to extract keypoints and hand region crops, respectively. Such necessities add computational overhead, decreasing inference speed and limiting the model's real-time usability, particularly on devices with limited processing power. There is an opportunity to replace an additional model with center cropping augmentation in the HandReader$_{RGB}$, provided the input focuses on the hands' region in the center of the frame. The HandReader$_{RGB+KP}$ is subject to the limits of other HandReader architectures, but it can utilize only the keypoint prediction model and obtain crops operating received points.

References

Yutong Chen, Ronglai Zuo, Fangyun Wei, Yu Wu, Shujie Liu, and Brian Mak. 2022. Two-stream network for sign language recognition and translation. In *Advances in Neural Information Processing Systems*, volume 35.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learn-

Experiment	Packed sequence	RNN	Count shift	Shift type	Image Augs.	BS	Letter accuracy
best	✓	**GRU**	✓	TSAM	✓	**4**	**72.0**
Packed sequence	✗	GRU	✗	TSM	✓	4	$70.6_{-1.4}$
RNN	✓	LSTM	✓	TSAM	✓	4	$70.31_{-1.69}$
	✓	✗	✓	TSAM	✓	4	$61.0_{-11.0}$
Count shift	✓	GRU	✗	TSAM	✓	4	$70.51_{-1.49}$
Image Augs.	✓	GRU	✓	TSAM	✗	4	$70.49_{-1.51}$
BS	✓	GRU	✓	TSAM	✓	2	$71.46_{-0.54}$
	✓	GRU	✓	TSAM	✓	6	$70.1_{-1.9}$
Baseline*	✗	GRU	✗	None	✓	4	$69.63_{-2.37}$

Table 7: Ablation study of HandReader$_{RGB}$ components. * denotes the baseline setup with a ResNet34 configuration (without TSM or TSAM shift operations), which already outperforms previous state-of-the-art solutions. Here, we define BS as batch size and RNN as the type of recurrent neural network used, which may include LSTM, GRU, or a linear layer without recurrence. "Packed sequence" refers to a modification of the TSM mechanism that enables processing each video independently during the shift operation, as described in Section 3.1.

Experiment	Keypoints			RNN	Conv. modules	Image Augs.	BS	Letter accuracy
	hand	pose	face					
best	✓	✓	✗	**GRU**	**1**	✓	**4**	**69.31**
Keypoints	✓	✗	✗	GRU	1	✓	4	$68.54_{-0.77}$
	✓	✓	✓			✓		$69.08_{-0.23}$
	✓	✗	✓			✓		$67.59_{-1.72}$
	✗	✓	✓			✓		$10.02_{-59.29}$
	✗	✓	✗			✓		$10.36_{-58.95}$
	✗	✗	✓			✓		$11.08_{-58.23}$
RNN	✓	✓	✗	LSTM	1	✓	4	$67.79_{-1.52}$
	✓	✓	✗	✗		✓		$63.88_{-5.43}$
Conv. modules	✓	✓	✗	GRU	2	✓	4	$68.09_{-1.22}$
	✓	✓	✗		3	✓		$68.11_{-1.2}$
Image Augs	✓	✓	✗	GRU	1	✗	4	$62.43_{-6.88}$
BS	✓	✓	✗	GRU	1	✓	2	$69.04_{-0.27}$
	✓	✓	✗			✓	6	$69.26_{-0.05}$
	✓	✓	✗			✓	8	$68.45_{-0.86}$

Table 8: Ablation study of HandReader$_{KP}$ components. Here, "Conv Modules" denotes the number of convolutional modules from the Conformer architecture that are used inside the encoder. Other terms are defined in Table 7.

ing phrase representations using rnn encoder-decoder for statistical machine translation. *Preprint*, arXiv:1406.1078.

Ashley Chow, Glenn Cameron, Manfred Georg, Mark Sherwood, Phil Culliton, Sam Sepah, Sohier Dane, and Thad Starner. 2023. Google – american sign language fingerspelling recognition. https://kaggle.com/competitions/asl-fingerspelling. Kaggle.

Pooya Fayyazsanavi, Negar Nejatishahidin, and Jana Košecká. 2024. Fingerspelling posenet: Enhancing fingerspelling translation with pose-based transformer models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1120–1130.

Globo Language Solutions. 2023. Why are asl interpreters so hard to find? https://www.helloglobo.com/blog/why-are-asl-interpreters-so-hard-to-find.

Alex Graves. 2012. Connectionist temporal classification. *Supervised sequence labelling with recurrent neural networks*, pages 61–93.

Shester Gueuwou, Xiaodan Du, Greg Shakhnarovich, and Karen Livescu. 2025a. SignMusketeers: An efficient multi-stream approach for sign language translation at scale. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 22506–22521.

Shester Gueuwou, Xiaodan Du, Greg Shakhnarovich, Karen Livescu, and Alexander H. Liu. 2025b. SHuBERT: Self-supervised sign language representation learning via multi-stream cluster prediction. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 28792–28810.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. 2020. Conformer: Convolution-augmented

transformer for speech recognition. *Preprint*, arXiv:2005.08100.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9:1735–80.

Byeongkeun Kang, Subarna Tripathi, and Truong Q Nguyen. 2015. Real-time sign language fingerspelling recognition using convolutional neural networks from depth map. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 136–140. IEEE.

Wuttipong Kumwilaisak, Peerawat Pannattee, Chatchawarn Hansakunbuntheung, Nattanun Thatphithakkul, et al. 2022. American sign language fingerspelling recognition in the wild with iterative language model construction. *APSIPA Transactions on Signal and Information Processing*, 11(1).

Ji Lin, Chuang Gan, and Song Han. 2019. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7083–7093.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *Preprint*, arXiv:1711.05101.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. 2019. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*.

National Deaf Center. 2022. The asl interpreter shortage and its impact on accessibility in college settings. https://nationaldeafcenter.org/news-items/the-asl-interpreter-shortage-and-its-impact-on-accessibility-in-college-settings/.

Nimdzi Insights. 2021. American sign language interpreting: The size and state of the market. https://www.nimdzi.com/asl-interpreting/.

Peerawat Pannattee, Wuttipong Kumwilaisak, Chatchawarn Hansakunbuntheung, Nattanun Thatphithakkul, and C-C Jay Kuo. 2024. American sign language fingerspelling recognition in the wild with spatio temporal feature extraction and multitask learning. *Expert Systems with Applications*, 243:122901.

Maria Parelli, Katerina Papadimitriou, Gerasimos Potamianos, Georgios Pavlakos, and Petros Maragos. 2020. Exploiting 3d hand pose estimation in deep learning-based sign language recognition from rgb videos. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 249–263. Springer.

Bowen Shi. 2023. Toward american sign language processing in the real world: Data, tasks, and methods. *Preprint*, arXiv:2308.12419.

Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Jonathan Michaux, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2018. American sign language fingerspelling recognition in the wild. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 145–152.

Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2019. Fingerspelling recognition in the wild with iterative visual attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Hao Zhou, Wengang Zhou, Yun Zhou, and Houqiang Li. 2021. Spatial-temporal multi-cue network for sign language recognition and translation. *IEEE Transactions on Multimedia*, 24:768–779.

Znaki: A Russian Fingerspelling Recognition Dataset

Petr Surovtsev[1]
pysurovtsev@daimld.tech

Pavel Korotaev[1]
pkorotaevdm@gmail.com

Alexander Nagaev[1]
aonagaev@daimld.tech

Alexander Kapitanov[1]
aakapitanov@daimld.tech

[1]Department of Artificial Intelligence and Machine Learning Development, DAIMLD

Abstract

Fingerspelling — the sequential signing of letters that compose proper names, technical terms, and out-of-vocabulary words — is an integral component of any sign language. Despite Russian Sign Language (RSL) being used by more than 150,000 native signers, no public fingerspelling dataset for RSL exists, in contrast to American, British, German, and several other sign languages. We introduce Znaki[1], the first open RSL fingerspelling dataset: 37,252 video recordings (35.4 hours) of 1,593 unique sequences — proper names sampled from 16 domain categories — recorded by 68 native RSL signers (37 Deaf, 29 hard-of-hearing, and 2 certified interpreters) through a crowdsourcing pipeline with a two-stage signer qualification procedure and expert validation. The dataset covers all 33 Cyrillic handshapes, and its letter-frequency distribution matches the Russian National Corpus. As reference points for future research, we establish two baselines adapted from American Sign Language fingerspelling recognition: a Squeezeformer-based encoder-decoder, reaching a character-level accuracy ($1 - \mathrm{CER}$) of 0.932 and a sequence accuracy of 0.596, and Fingerspelling-PoseNet, reaching 0.886 and 0.545, respectively.

1 Introduction

Sign languages are full-fledged natural languages that exhibit cultural and regional variation. There are approximately 72 million Deaf and hard-of-hearing people worldwide, and more than 300 sign languages[2]. Sign language is typically the first (native) language of Deaf individuals, and without adequate education, it may remain their only language — which can hinder communication and limit access to certain products or services. Given the limited number of sign language interpreters, automatic sign language recognition systems can help reduce communication barriers and contribute to preserving the culture of these languages. As of 2024, there are over 150,000 Deaf signers of Russian Sign Language (RSL) in Russia[3]. The use of RSL also extends to the countries of the former Soviet Union (now the CIS countries), possibly with minor regional variations, which implies significantly wider use of RSL in the world.

Sign language comprises manual and non-manual components. Signs can be classified based on their articulation into one-handed and two-handed signs, with two-handed signs being either symmetrical (both hands perform identical movements) or asymmetrical (one hand performs the primary movement while the other serves a supporting role). In RSL, each of the four manual components of a sign — handshape, location, orientation, and movement — contributes to the meaning of the sign. In RSL, fingerspelling is primarily used to represent proper names, technical and specialized terms, words for which no conventional sign exists, as well as words that are conventionally fingerspelled — such as “зал” (*zal*, “hall”) and “как” (*kak*, “how”). The ability to recognize fingerspelling — an integral part of sign language — is an important step toward full sign language recognition.

The Russian language has 33 letters, which correspond to 33 unique fingerspelling signs (handshapes) in RSL (see Figure 1). Handshapes are produced with a single dominant hand, approximately at chest level, with each letter being clearly articulated. Fingerspelling recognition is complicated by the presence of dynamic signs (“Д”, “Ё”, “З”, “Й”, “Ц”, “Щ”, “Ъ”, “Ь” — *D, Yo, Z, Y, Ts, Shch, hard sign, soft sign*) and signs with similar hand configurations — such as “Ш” (*Sh*) and “Щ” (*Shch*), “Е” (*E*) and “Ё” (*Yo*), “И” (*I*) and “Й” (*Y*), and

[1]https://github.com/ai-forever/HandReader

[2]https://education.nationalgeographic.org/resource/sign-language/

[3]https://voginfo.ru/about/

Proceedings of the Second Workshop on Sign Language Processing, pages 203–212
October 29, 2026 ©2026 Association for Computational Linguistics

“P” (*R*) and “H” (*N*).

This paper presents an efficient data collection methodology and a new fingerspelling dataset for Russian Sign Language recognition. The main contributions of this paper are as follows:

- Znaki, the first publicly available fingerspelling dataset for RSL, comprising 37,252 videos (35.4 hours) of 1,593 unique proper names recorded by 68 native RSL signers.
- A Deaf-community-informed phrase sampling covering all 33 Cyrillic handshapes and 696 letter bigrams, with a letter-frequency distribution matching the Russian National Corpus.
- A two-stage signer qualification procedure — an RSL proficiency exam followed by a fingerspelling distractor test.
- Baseline results from two pose-based fingerspelling recognition architectures previously developed for American Sign Language (ASL).

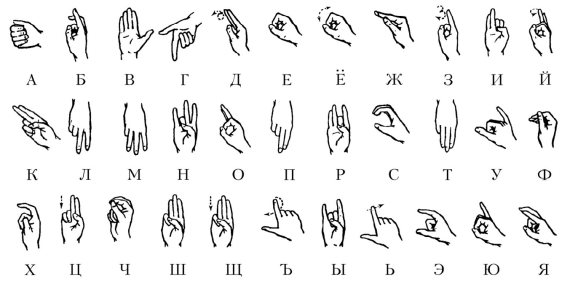


Figure 1: The 33 signs of the RSL alphabet, each corresponding to a letter of the Russian alphabet. Arrows indicate the direction of movement for 8 dynamic signs.

2 Related Work

Sign language recognition has been an active area of research for many years. Early approaches primarily focused on image-based recognition of static hand signs in ASL (Barczak et al., 2011; Kang et al., 2015), RSL (Makarov et al., 2019), BdSL (Podder et al., 2020), and other sign languages (Szyszka, 2021; Kenshimov et al., 2021). Nevertheless, dynamic signs are present in virtually all sign languages, and recognizing them necessitates capturing the temporal dimension of the sign. Robust fingerspelling recognition therefore requires a dataset comprising video samples. At present, two categories of fingerspelling data are publicly available: isolated handshapes (individual letters) and continuous fingerspelling (complete words or phrases). Table 1 summarizes the publicly available resources most relevant to our work.

As noted in Section 1, no continuous fingerspelling datasets exist for RSL.

2.1 Isolated Handshape Datasets

One of the earliest fingerspelling datasets, the RWTH German Fingerspelling Database (Dreuw et al., 2006), contains 1,400 video sequences (320×240 and 352×288 resolution, 25 fps) representing 35 DGS signs recorded from 20 participants across two sessions. Each sign was captured simultaneously by two cameras from different viewpoints under varying lighting and background conditions. Participants wore no special clothing and received no training, making the dataset more representative for unconstrained recognition systems.

The Nepali Sign Language dataset (NSL23) (Sunuwar et al., 2023) is the first dataset for NSL, containing 630 videos (1,205 signs) representing 36 consonants and 13 vowels. Recordings were collected from 14 volunteers (9 beginners, 5 experts) under various conditions (normal lighting, low lighting, prepared/unprepared, and real-world environments).

For RSL handshapes, two datasets are known: the publicly available Bukva dataset (Kvanchiani et al., 2024) and the private NSTU dataset (Grif and Kondratenko, 2021). The Bukva dataset comprises 3,757 videos covering all 33 letters of the Russian alphabet, with an average of 113 videos per class (49 frames per video). Recorded by 155 signers — including Deaf individuals, RSL learners, and teachers — the dataset represents a wide age range (9 to 81 years). The NSTU dataset comprises 13,412 images and videos of 33 Russian fingerspelling signs recorded from four signers (waist-up, right hand at chest level). Due to the limited signer diversity, the authors augmented the dataset with an additional 1,566 samples collected from YouTube and custom recordings to improve real-world performance.

Moreover, general sign language datasets commonly include a collection of handshapes as well (LI et al., 2020; Kapitanov et al., 2023).

2.2 Continuous Fingerspelling Datasets

Among the first large-scale ASL fingerspelling collections are the ChicagoFSWild (Shi et al., 2019b) and ChicagoFSWild+ (Shi et al., 2019a) datasets.

Dataset	Language	Samples	Signers	Hours	Type
RWTH (Dreuw et al., 2006)	DGS	1,400	20	—	isolated
NSL23 (Sunuwar et al., 2023)	NSL	630	14	—	isolated
Bukva (Kvanchiani et al., 2024)	RSL	3,757	155	—	isolated
NSTU (Grif and Kondratenko, 2021)	RSL	14,978	4	—	isolated (private)
ChicagoFSWild (Shi et al., 2019b)	ASL	7,304	168	—	continuous
ChicagoFSWild+ (Shi et al., 2019a)	ASL	55,232	260	—	continuous
FSboard (Georg et al., 2024)	ASL	151,000	147	266	continuous
FS23K (Chan et al., 2026)	BSL	23,000	27	10.5	continuous
BANZ-FS (Shen et al., 2026)	BANZSL	35,028	116	—	continuous
Znaki (ours)	RSL	37,252	68	35.4	continuous

Table 1: Publicly available fingerspelling datasets. Znaki is the first continuous fingerspelling dataset for RSL.

ChicagoFSWild was collected from online sources (YouTube, ASLized, and deafvideo.tv). The authors collected 214 raw ASL videos and manually annotated all fingerspelling clips, resulting in 7,304 sequences from 168 signers. The dataset includes proper nouns, other words, and abbreviations. Like ChicagoFSWild, the ChicagoFSWild+ dataset was also collected from online sources; however, annotation of the fingerspelling sequences was performed through a crowdsourcing platform. ChicagoFSWild+ contains 55,232 sequences from 260 signers in total, split into 50,402 training sequences (216 signers), 3,115 development sequences (22 signers), and 1,715 test sequences (22 signers).

The FSboard dataset (Georg et al., 2024) contains 151,000 sequences with an average length of 21 characters, totaling 3.2 million characters and 266 hours of video — recorded entirely on smartphones — making it the largest previously available ASL fingerspelling dataset. It includes 147 unique signers split into training (117), validation (15), and test (15) sets with no phrase overlap between splits. FSboard includes numbers alongside fingerspelling and covers five phrase domains: MacKenzie phrases, random URLs, random street addresses, random names, and random phone numbers.

The FS23K dataset (Chan et al., 2026) contains 23K BSL fingerspelling instances with letter-level annotations and 133K temporal boundaries. It comprises 23K words, 139K letter annotations, 920K frames, 10.5 hours of video, and 27 signers. The letter distribution closely follows natural frequencies from BBC broadcast data.

The most recent resource, BANZ-FS (Shen et al., 2026), is the first large-scale dataset of two-handed fingerspelling in BANZSL — the manual alphabet shared by British, Australian, and New Zealand Sign Languages. It comprises 35,028 fingerspelling instances (over 200K fingerspelled characters) from 116 signers, compiled from three heterogeneous sources: sign-interpreted news broadcasts, controlled laboratory recordings, and web vlogs. Each instance carries multi-level alignment (video to subtitles, fingerspelled letters, and target lexicons), and the dataset supports four benchmark tasks: isolated fingerspelling recognition, fingerspelling detection, detection-then-recognition, and fingerspelling recognition in the context of continuous signing.

Znaki differs from these recent resources in both language and collection methodology. FSboard, FS23K, and BANZ-FS cover ASL and BANZSL — with BANZ-FS specifically targeting the two-handed BANZSL alphabet — whereas Znaki addresses the one-handed manual alphabet of RSL, for which no continuous fingerspelling data was previously available. Furthermore, unlike datasets mined from broadcast or web footage, all Znaki recordings were collected specifically for the dataset through a crowdsourcing pipeline with a two-stage signer qualification procedure and subsequent expert validation, providing explicit control over signer proficiency, phrase coverage, and annotation quality.

3 Dataset Collection

Sign language datasets are typically collected using one of three common approaches: (1) studio recordings, (2) recordings sourced from the internet or television broadcasts, and (3) crowdsourcing plat-

forms. The first method suffers from several limitations: a small number of participants and limited clothing diversity; limited variation in backgrounds and lighting; low diversity of recording devices; and overly idealized signing resulting from expert supervision during recording. The limitations of the second approach include a limited number of signers and, most critically, the necessity of additional data filtering, processing, and annotation. The third method avoids many of the limitations listed above; however, it requires a well-established data collection pipeline involving expert validation procedures. The Znaki dataset was collected using the crowdsourcing platform TagMe[4]. Figure 2 provides samples from the Znaki dataset. The data collection pipeline consisted of the following stages: signer recruitment, video collection, video validation, video processing, and video annotation.



Figure 2: Sample frames from the Znaki dataset.

3.1 RSL Experts

RSL experts included Deaf or hard-of-hearing individuals and certified interpreters with relevant qualifications. The majority were members of the VOG (All-Russian Society of the Deaf)[5] community, including RSL researchers and teachers. These experts were responsible for screening participants during the project admission stages and were also granted access to video verification and annotation tasks.

3.2 Phrase Selection

Since fingerspelling is primarily used for signing proper names, only words and phrases belonging to this group were selected for the dataset. We consulted with Deaf experts and identified the 16 most common categories of proper names (see Table 2), selecting approximately 100 well-known words and phrases for each category. The proper names for each category were generated using GPT-4o (OpenAI, 2024), a large language model (LLM) developed by OpenAI (the prompt template is provided in Appendix A). All generated words and phrases were subsequently verified by RSL experts and manually refined when necessary. The resulting phrases contain between 1 and 4 proper names, with an average of 1.4 words ($\sigma = 0.6$), and range from 3 to 29 (excluding the space character) characters in length, with an average of 10,0 characters ($\sigma = 4.1$).

Category	Count
Moscow Metro	231
Social terms	121
Banking terms	107
Countries	99
Russian cities	99
Authors	98
Book titles	97
Government institutions	96
World cities	93
Movie titles	92
World landmarks	91
Disease names	90
Russian landmarks	84
Professions	75
Geographical names	70
Personal names	50

Table 2: The number of unique samples in each category of the Znaki dataset.

3.3 Signer Selection

We specifically sought to recruit Deaf or hard-of-hearing signers who are members of the VOG community. However, the platform did not restrict projects to specific user groups — they were accessible to all annotators. To ensure data quality, we implemented a two-stage screening process.

RSL knowledge exam. Candidates were required to pass a knowledge test of RSL, which involved recording a video translating Russian sentences into RSL. RSL experts and RSL teachers evaluated the correctness of the submissions and determined whether candidates would proceed to the next test. This exam made it possible to assess the candidates' RSL proficiency and their ability to understand and follow task instructions.

[4] https://tagme.sberdevices.ru
[5] https://voginfo.ru/

Fingerspelling exam. Candidates who passed the RSL exam were then required to complete an additional fingerspelling test. This test consisted of 15 tasks. In each task, candidates watched a video of RSL fingerspelling and had to select the correct Russian translation from a list of words, with a time limit of 30 seconds per task. The videos were recorded by RSL experts, who also created the incorrect response options. These experts deliberately designed plausible distractors that could be mistaken for the correct answer due to a candidate's lack of knowledge or attention. A score of at least 85% correct answers was required to pass. 68 candidates out of 100 were accepted to the main tasks.

3.4 Video Collection

The video collection task was accompanied by written instructions describing the recording rules and quality requirements. Additionally, signers were provided with a video in which RSL teachers explained the same instructions in sign language.

The recording rules were as follows: (1) only one person could appear in the video; (2) the signing hand had to remain fully within the frame; and (3) camera shaking and visual effects were not permitted. The platform verification process accepted only videos with a minimum HD resolution, a duration exceeding one second, and a frame rate of at least 15 fps. Duplicate videos were rejected, and signers were allowed to re-record submissions.

Signers could record videos in arbitrary environments, making the dataset heterogeneous with respect to recording conditions. To further increase data diversity, a daily limit of 30 tasks per signer was enforced. Figure 3 illustrates an example of a video recording task.



Figure 3: Task for recording the proper name «Виктор Гюго» (Victor Hugo).

3.5 Video Validation

Each video was checked for the correctness of the translation from Russian into RSL fingerspelling sequences. The videos were validated by a diverse group of participants, including Deaf and hard-of-hearing individuals, certified RSL interpreters, and RSL teachers and educators, all of whom confirmed their proficiency in Russian Sign Language. Validators also verified that each video complied with the recording requirements.

To become a validator, crowdworkers were required to complete the mandatory qualifying tasks described in Section 3.3, undergo training, and pass a video validation exam.

Each video was reviewed by at least three validators. In cases of insufficient inter-annotator agreement, additional validators were recruited until a satisfactory level of consensus was achieved. Overall, 82% of the samples received unanimous validator agreement, i.e., inter-annotator agreement of the validation judgments, indicating a high level of annotation reliability. On average, validating a single video took 17.3 seconds.

Approximately 12% of the collected videos were rejected during expert validation due to signing errors or failure to comply with the recording requirements and instructions.

3.6 Time Interval Annotation

Time interval annotation was required to separate the sign sequence from other activity periods (see Figure 4). As in the validation stage, crowdworkers received training and passed an exam to gain access to the annotation tasks. Validated videos were processed to 30 frames per second (fps) to ensure consistent time alignment. Each video was annotated by three users, and the annotations were subsequently aggregated into a single result using an open-source aggregation framework[6]; 95% of the videos achieved an agreement level above 80%. This annotation approach made it possible to determine the fingerspelling speed, with an average rate of 3.2 signs (letters) per second. Importantly, the signing rate was not strongly dependent on the sequence length, as the standard deviation was only 0.1 signs per second.

Note that the annotations mark the boundaries of the entire fingerspelled sequence (start and end); the dataset does not include frame-level or letter-level segmentation. Reliable letter boundaries are

[6] https://github.com/ai-forever/aggme

difficult and costly to annotate at scale, since adjacent handshapes blend into one another due to coarticulation (cf. FS23K (Chan et al., 2026), where letter boundaries were annotated for a substantially smaller collection). We discuss the implications of this design choice in the Limitations section.

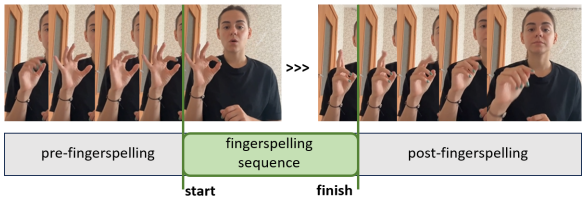


Figure 4: Illustration of the annotation scheme for fingerspelling sequence boundaries (start and end).

3.7 Dataset Characteristics

Signers. The dataset was recorded by 68 signers, including 37 Deaf individuals, 29 hard-of-hearing individuals, and two sign language interpreters (see Figure 5). The signers are aged from 20 to 54 years, with 52 women and 16 men.

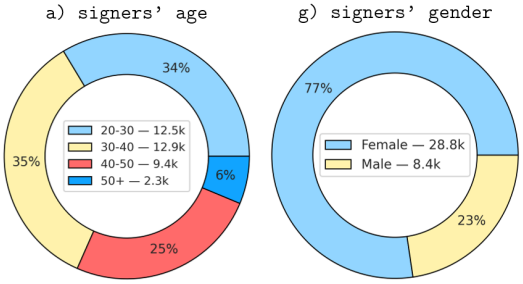


Figure 5: Age and gender distribution in the Znaki dataset.

Sequences. The dataset contains 1,593 unique proper names, covering all 33 letters of the Russian alphabet and 696 letter bigrams (≈64% of all possible combinations). The letter-frequency distribution in Znaki corresponds to that of the Russian National Corpus[7] (see Figure 6).

Videos. Znaki contains 37,252 videos (35.4 hours in total) with a minimum frame side of 720 pixels, trimmed according to the time interval annotations, resulting in about 42 GB of data. Each sequence was recorded by between 8 and 41 different signers, averaging 23 videos per phrase. The average length of each sequence is 102 frames (see Figure 7). More than 80% of the dataset's videos are

[7] https://ruscorpora.ru/en/

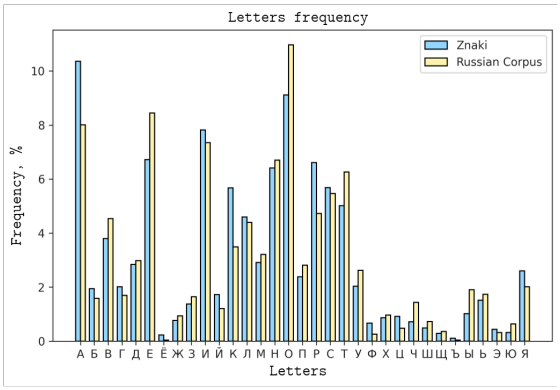


Figure 6: Comparison of letter frequencies in the Znaki dataset and the Russian National Corpus.

in Full HD format, and the rest are in HD. About 88% of the samples were recorded on a phone camera, and the rest on a laptop.

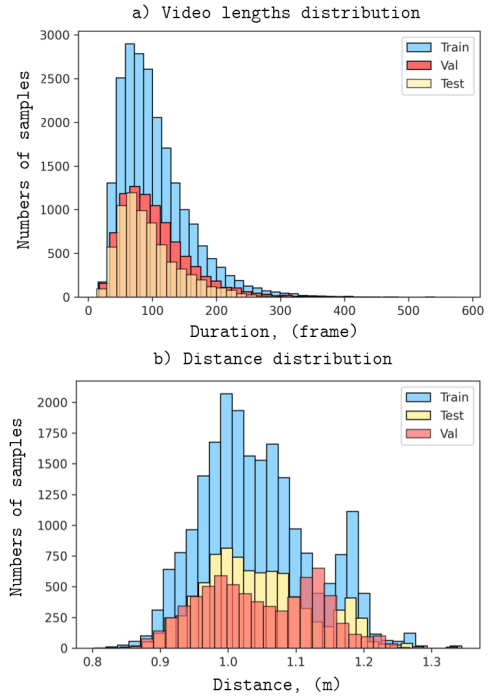


Figure 7: Znaki dataset statistics: a) video length distribution; b) distance distribution, where the distance to the camera is approximately estimated in meters by computing the length between the signer's left and right shoulders using MediaPipe (Lugaresi et al., 2019).

Split. The initial collection comprised more than 50,000 videos. To avoid potential data leakage — specifically, to ensure no overlap of phrases or signers between training, validation, and test sets — a portion of the data was deliberately excluded from the final dataset. We split the dataset into training (58%), validation (23%), and test (19%) sets, each

containing unique phrases only: 743, 300, and 550, respectively. No signers from the test set overlap with the training and validation sets. Additionally, we ensured a proportional distribution of letters across the splits (see Figure 8).

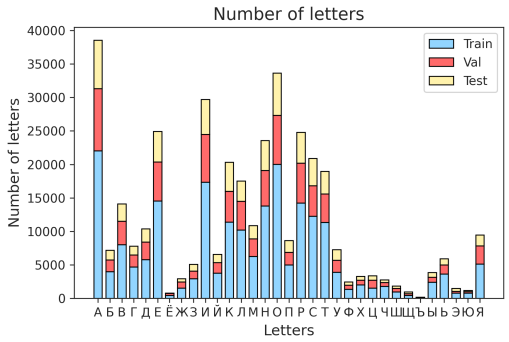


Figure 8: Distribution of the total number of letters across the dataset splits.

4 Experiments

4.1 Experimental Setup

The goal of our experiments is not to introduce a new recognition architecture, but to validate the proposed dataset as a viable benchmark for RSL fingerspelling recognition. Since no publicly available benchmark for this task currently exists, we establish baseline results using two publicly available fingerspelling recognition models (Henkel, 2023; Fayyazsanavi et al., 2023) originally developed for ASL.

All experiments were conducted on a single H100 80 GB GPU with 12 CPU cores and 244 GB RAM. Training each model required approximately 14 GPU hours. Training hyperparameters are listed in Appendix B.

Dataset Split. No additional preprocessing or modifications were applied to the proposed train/validation/test split. The models were trained on the training set, validated on the validation set, and evaluated on the test set.

Input Representation. We used MediaPipe Holistic (Grishchenko and Bazarevsky, 2020) to extract a total of 543 landmarks per frame: 33 pose landmarks, 21 landmarks for each hand, and 468 facial landmarks.

Evaluation Metrics. We evaluate our models using two complementary metrics: Character Error Rate (CER) and Accuracy (Acc). CER measures the edit distance between the predicted and ground-truth character sequences. To improve interpretability (higher is better), we report $1 - \mathrm{CER}$, where a value of 1.0 corresponds to perfect character-level prediction:

$$1 - \mathrm{CER} = 1 - \frac{S + D + I}{N},$$

where S is the number of substitutions, D is the number of deletions, I is the number of insertions, and N is the total number of characters in the ground-truth sequence.

Accuracy is defined as the proportion of completely correct sequence predictions (all characters in the correct order) relative to the total number of test samples:

$$\mathrm{Acc} = \frac{\text{\# fully correct sequences}}{\text{\# total sequences}}.$$

4.2 Baseline Models

We benchmark the dataset using two publicly available architectures previously proposed for ASL fingerspelling recognition.

Transformer-Based. Our first baseline is based on the winning solution of the Kaggle ASL Fingerspelling competition (Henkel, 2023). The architecture employs an improved Squeezeformer encoder adapted for MediaPipe landmark sequences together with a two-layer transformer decoder for sequence prediction. From the full set of extracted landmarks, the model uses a subset of 130 per frame: 42 hand landmarks (21 per hand), 12 pose landmarks, and 76 facial landmarks.

Fingerspelling-PoseNet. Our second baseline is based on Fingerspelling-PoseNet (Fayyazsanavi et al., 2023), a pose-based transformer encoder-decoder architecture designed for fingerspelling recognition from hand pose sequences. The model operates on the 21 2D landmarks of the automatically detected signing hand and performs autoregressive sequence prediction.

Adaptation to RSL. Both baselines were transferred to RSL with minimal changes. The only modification was a configuration-level replacement of the output vocabulary with the 33 letters of the Cyrillic alphabet; the source code of both public implementations was left unchanged, so the input representation, normalization, architecture, optimizer settings, learning rate schedule, and augmentations remain the defaults released by the original authors (see Appendix B).

4.3 Results

We evaluate the two baseline architectures on the held-out test split of the Znaki dataset. Table 3 reports character-level $1 - \mathrm{CER}$ and phrase-level (Acc) performance.

Model	$1 - \mathrm{CER} \uparrow$	Acc $\uparrow$
FS-PoseNet	0.886	0.545
Transformer-Based	**0.932**	**0.596**

Table 3: Baseline results on the proposed RSL fingerspelling dataset.

The Transformer-Based model, adapted from ASL to RSL, achieves strong character-level accuracy ($1 - \mathrm{CER} = 0.932$) but exhibits a substantial gap at the sequence level (Acc = 0.596), indicating compounding errors in sequence prediction.

Error analysis reveals three primary failure modes:

- **Visual confusability**: high confusion rates among visually similar handshapes (e.g., "Ш"/"Щ" (*Sh/Shch*), "Е"/"Ё" (*E/Yo*), "И"/"Й" (*I/Y*), "Р"/"Н" (*R/N*)), accounting for 38% of character substitutions.
- **Dynamic sign instability**: temporal segmentation errors for movement-based handshapes ("Д", "З", "Й", "Ц", "Щ" — *D, Z, Y, Ts, Shch*), leading to 27% of deletions and insertions.
- **Length-dependent degradation**: Acc decreases by 4.2% per additional character for sequences of $\geq$4 letters, reflecting accumulated alignment uncertainty.

Despite heterogeneous recording conditions (lighting, viewpoint, device), models trained on MediaPipe Holistic landmarks (Grishchenko and Bazarevsky, 2020) demonstrate robust generalization: no performance drop is observed when evaluating on signers unseen during training, confirming the absence of signer-specific overfitting.

These baselines establish Znaki as a challenging benchmark for RSL fingerspelling recognition and provide a reference point for future methodological advances.

5 Conclusion

In this work, we presented Znaki — the first public fingerspelling dataset for RSL and the largest RSL fingerspelling resource to date. By involving only native and near-native members of the signing community, Znaki captures natural intra- and inter-signer variability while ensuring the linguistic authenticity of the collected data, addressing a critical gap: prior to our work, no publicly available RSL fingerspelling datasets existed.

We involved RSL experts in defining the categories of lexical sequences and used LLMs to generate candidate proper names for recording. In addition, we developed a participant verification procedure for the crowdsourcing platform to improve recording quality and annotation reliability.

To validate the usability of the proposed dataset, we trained two publicly available strong baselines previously developed for ASL fingerspelling recognition. The successful transfer of these architectures to RSL fingerspelling demonstrates the suitability of the proposed dataset for training and evaluating modern fingerspelling recognition models.

We believe that publicly releasing Znaki will stimulate further progress in sign language processing, particularly for under-resourced sign languages.

Limitations

The Znaki dataset reflects the demographic composition of its participants and therefore exhibits several imbalances. The dataset was collected from signers aged 20–54 years, without representation of children or elderly signers, whose signing patterns may differ substantially. In addition, the gender distribution is imbalanced (52 women and 16 men), and all participants belong to a relatively homogeneous ethnic and geographic background. These factors may limit the generalization of models trained on Znaki to broader signer populations.

The dataset focuses exclusively on fingerspelled proper names and predefined phrase categories. As a result, it does not cover several important aspects of real-world fingerspelling communication, including numerical sequences, conversational phrases, abbreviations, partial fingerspelling, fingerspelling embedded in mixed lexical contexts (interleaved with lexical signs), and coarticulation between fingerspelling and surrounding signs in continuous signing. Although the selected phrase categories were designed in consultation with Deaf experts, the phrase distribution may not fully reflect natural fingerspelling usage frequencies in the Deaf community. More broadly, this scope is deliberate: Znaki is a fingerspelling dataset rather than a full

continuous sign language recognition or translation resource, and it does not include RSL gloss annotations — fingerspelling is one component of sign language communication, not a substitute for full sign language recognition.

The annotations provide only sequence-level temporal boundaries (start and end of fingerspelling); frame-level and letter-level segmentation is not available. This limits the direct use of Znaki for research on letter-to-frame alignment, fine-grained temporal modeling of handshape transitions, and fingerspelling detection within continuous signing. Adding letter-level temporal annotations is a natural direction for future extensions of the dataset.

The recording setup also introduces several constraints. Most videos were recorded using smartphone cameras in frontal-view settings, with limited variation in viewpoint. The dataset does not explicitly cover challenging scenarios such as severe occlusions, extreme lighting conditions, or multi-view recordings.

Our experimental evaluation is limited to two publicly available baseline architectures originally developed for ASL fingerspelling recognition. Therefore, the reported results should be viewed as initial baselines rather than exhaustive performance estimates for the dataset.

Ethical Statement

The legal and ethical aspects of the dataset collection procedure were reviewed and approved by our institution's legal team. All crowdworkers provided informed consent through the crowdsourcing platform, authorizing the processing and public release of their data for research purposes. Participation in the project was voluntary, and all contributors received financial compensation proportional to the time required for task completion. Compensation rates were aligned with the average income levels in the Russian Federation.

To protect participant privacy, all contributors were identified only through anonymized hash-based user identifiers. Since facial expressions and non-manual components constitute an essential part of sign language communication, facial information was retained in the recordings and may be misused in surveillance-related applications.

To mitigate such risks, the dataset is released exclusively for research purposes under a restricted-use license.

References

Andre L. C. Barczak, Napoleon H. Reyes, Mary E Abastillas, A. Piccio, and Teo Susnjak. 2011. A new 2d static hand gesture colour image dataset for asl gestures.

Alyssa Chan, Taein Kwon, and Andrew Zisserman. 2026. Recognising bsl fingerspelling in continuous signing sequences. *Preprint*, arXiv:2603.19523.

Philippe Dreuw, Thomas Deselaers, Daniel Keysers, and Hermann Ney. 2006. Modeling image variability in appearance-based gesture recognition. In *ECCV Workshop on Statistical Methods in Multi-Image and Video Processing*, pages 7–18.

Pooya Fayyazsanavi, Negar Nejatishahidin, and Jana Kosecka. 2023. Fingerspelling posenet: Enhancing fingerspelling translation with pose-based transformer models. *Preprint*, arXiv:2311.12128.

Manfred Georg, Garrett Tanzer, Saad Hassan, Maximus Shengelia, Esha Uboweja, Sam Sepah, Sean Forbes, and Thad Starner. 2024. Fsboard: Over 3 million characters of asl fingerspelling collected via smartphones. *Preprint*, arXiv:2407.15806.

M. G. Grif and Y. K. Kondratenko. 2021. Development of a software module for recognizing the fingerspelling of the russian sign language based on lstm. *Journal of Physics: Conference Series*, (1).

Ivan Grishchenko and Valentin Bazarevsky. 2020. Mediapipe holistic — simultaneous face, hand and pose prediction, on device. Google MediaPipe.

Christof Henkel. 2023. Kaggle asl fingerspelling recognition: 1st place solution. `https://github.com/ChristofHenkel/kaggle-asl-fingerspelling-1st-place-solution`.

Byeongkeun Kang, Subarna Tripathi, and Truong Q. Nguyen. 2015. Real-time sign language fingerspelling recognition using convolutional neural networks from depth map. *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 136–140.

Alexander Kapitanov, Kvanchiani Karina, Alexander Nagaev, and Petrova Elizaveta. 2023. Slovo: Russian sign language dataset. In *Computer Vision Systems: 14th International Conference, ICVS 2023, Vienna, Austria, September 27–29, 2023, Proceedings*, page 63–73.

Chingiz Kenshimov, Samat Mukhanov, Timur Merembayev, and Didar Yedilkhan. 2021. A comparison of convolutional neural networks for kazakh sign language recognition. *Eastern-European Journal of Enterprise Technologies*, pages 44–54.

Karina Kvanchiani, Petr Surovtsev, Alexander Nagaev, Elizaveta Petrova, and Alexander Kapitanov. 2024. Bukva: Russian sign language alphabet. *Preprint*, arXiv:2410.08675.

DONGXU LI, Cristian Rodriguez, Xin Yu, and HONGDONG LI. 2020. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, and 1 others. 2019. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*.

Ilya Makarov, Nikolay Veldyaykin, Maxim Chertkov, and Aleksei Pokoev. 2019. Russian sign language dactyl recognition. In *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*, pages 726–729.

OpenAI. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Kanchon Kanti Podder, Muhammad Chowdhury, Zaid Mahbub, and Muhammad Kadir. 2020. Bangla sign language alphabet recognition using transfer learning based convolutional neural network. *The Bangladesh journal of scientific research*, 31-33:20–26.

Xin Shen, Yan Ke, Xinyu Wang, and Xin Yu. 2026. BANZ-FS: BANZSL fingerspelling dataset. In *The Fourteenth International Conference on Learning Representations (ICLR)*.

Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2019a. Fingerspelling recognition in the wild with iterative visual attention. *Preprint*, arXiv:1908.10546.

Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Jonathan Michaux, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2019b. American sign language fingerspelling recognition in the wild. *Preprint*, arXiv:1810.11438.

Jhuma Sunuwar, Samarjeet Borah, and Aditi Kharga. 2023. Nsl23 dataset for alphabets of nepali sign language.

Ewa Szyszka. 2021. Polish Sign Language Alphabet 100.

A Phrase Generation Prompt

Candidate proper names for each of the 16 categories (Table 2) were generated with GPT-4o using the following prompt template:

> *Compile a list of the 100 most popular proper names from the "{category}" domain.*

All generated candidates were subsequently verified by RSL experts and manually refined when necessary (see Section 3.2).

B Training Details

Both baselines were trained on the training split with early stopping on the validation split, using the default hyperparameters of the respective public implementations (Henkel, 2023; Fayyazsanavi et al., 2023). We did not modify the source code of either repository: the training procedure, optimizer settings, learning rate schedule, and augmentations were kept exactly as released by the original authors, so the exact values can be found in the corresponding public repositories. The only adaptation required for RSL was a configuration-level change of the output vocabulary, replacing the original character set with the 33 letters of the Cyrillic alphabet. Each model was trained for approximately 14 hours on a single H100 80 GB GPU.

TransAlign-SLT: Complementary Temporal and Semantic Supervision for Gloss-Free Sign Language Translation

Pallapu Mohan Krishna[1] *, **Uppala Sanjay Kumar**[2], **M. K. Bhuyan**[1,3], **Debanga Raj Neog**[1]

[1]Mehta Family School of Data Science and Artificial Intelligence
[2]Department of Mathematics [3]Department of Electronics and Electrical Engineering
Indian Institute of Technology Guwahati, Assam, India

{k.pallapu,k.uppala,mkb,dneog}@iitg.ac.in

Abstract

Gloss-free sign language translation (SLT) aims to translate sign videos directly into spoken-language sentences without relying on costly gloss annotations. However, when supervision is available only at the sentence level, the correspondence between visual dynamics and linguistic content remains weak. We introduce **TransAlign-SLT**, a lightweight fully gloss-free framework that derives complementary temporal and semantic supervision from the available translations. Auxiliary TranslationCTC provides coarse temporal regularization, while Masked Entropic Optimal Transport (MOT) aligns visual temporal features with contextualized word-level representations from a frozen, corpus-adapted BERT encoder. Since the semantic branch is used only during training, the inference pipeline remains lightweight and autoregressive. TransAlign-SLT achieves **23.54 BLEU-4 / 48.46 ROUGE-L** on PHOENIX14T and **16.39 BLEU-4 / 44.68 ROUGE-L** on CSL-Daily, with **45.6M** inference parameters and **123 GFLOPs**. Ablation studies show that temporal modeling, semantic alignment, and decayed TranslationCTC supervision provide complementary benefits. Overall, the results show that useful visual-language structure can be learned directly from translation supervision without glosses, pseudo-glosses, or a large pretrained language decoder at inference.

1 Introduction

Sign languages express meaning through coordinated hand movements, facial expressions, body posture, and other non-manual cues. Sign Language Translation (SLT) aims to translate continuous sign videos into spoken-language sentences, supporting communication between signed and spoken languages.

Many existing SLT systems use gloss annotations as an intermediate form of supervision (Camgoz et al., 2018, 2020). Glosses provide useful temporal and lexical structure, but producing them requires linguistic expertise and substantial annotation effort, which makes them difficult to scale across languages and domains. This limitation has encouraged growing interest in *gloss-free* SLT, where models learn directly from paired sign videos and sentence-level translations (Zhou et al., 2023; Fu et al., 2023; Chen et al., 2025). In this setting, however, the model must infer both temporal structure and visual-language correspondence from the relatively weak supervision provided by sentence-level translations.

Recent gloss-free methods address this difficulty in several ways, including gloss-inspired attention (Yin et al., 2023), visual-language pretraining (Zhou et al., 2023; Jiao et al., 2024), cross-modal contrastive learning (Fu et al., 2023; Chen et al., 2025), and the use of large pretrained language or multimodal models (Wong et al., 2024; Gong et al., 2024; Chen et al., 2024; Kim et al., 2025). While these approaches have improved translation performance, they may involve additional pretraining stages, auxiliary recognition models, pseudo-label generation, or large language components. This raises a natural question: *can useful temporal and semantic structure be learned directly from raw video-translation pairs without gloss supervision or a large pretrained language decoder at inference?*

To investigate this question, we introduce **TransAlign-SLT**, a fully gloss-free framework that combines complementary temporal and semantic supervision. The visual stream couples multi-scale temporal modeling with bidirectional sequence refinement. During training, **TranslationCTC** provides a coarse temporal-linguistic prior, while **Masked Entropic Optimal Transport (MOT)** encourages soft semantic correspondence between visual temporal features and contextualized word-level representations obtained from a

*Corresponding author: k.pallapu@iitg.ac.in

Proceedings of the Second Workshop on Sign Language Processing, pages 213–223
October 29, 2026 ©2026 Association for Computational Linguistics

frozen, corpus-adapted BERT encoder. To obtain word-level representations, WordPiece subwords belonging to the same lexical unit are pooled before transport alignment. Translation is then generated autoregressively using an attention-based GRU decoder, with optional post-hoc CTC rescoring at inference.

We evaluate TransAlign-SLT on PHOENIX14T and CSL-Daily and examine the contributions of temporal modeling, semantic supervision, TranslationCTC weighting, and decoding.

Our main contributions are:

- We introduce **TransAlign-SLT**, a fully gloss-free SLT framework trained directly from sign videos and spoken-language translations without gloss or pseudo-gloss supervision.
- We combine **TranslationCTC** for coarse temporal regularization with **MOT-based semantic alignment** using corpus-adapted contextualized word representations and subword-aware pooling.
- We evaluate the framework on PHOENIX14T and CSL-Daily and provide component-level analyses of temporal modeling, semantic supervision, TranslationCTC, and decoding, while retaining an inference pipeline that does not require a large pretrained language decoder.

The implementation is publicly available at `https://github.com/pallapumohankrishna/TransAlign-SLT`.

2 Related Work

2.1 Gloss-Supervised and Weakly Supervised SLT

Early SLT systems often relied on gloss annotations as an intermediate source of supervision (Camgoz et al., 2018, 2020). Later work improved temporal, multi-cue, and sequence modeling through stochastic transformers, spatial-temporal representations, cross-modal reranking, and stronger sequence architectures (Voskou et al., 2021; Zhou et al., 2022; Zhao et al., 2022; Chen et al., 2022a,b; Zhang et al., 2023; Zhou et al., 2021). While these approaches have been effective, many of them depend on gloss annotations, which require substantial linguistic expertise and are difficult to scale across languages and domains.

Weakly supervised methods aim to reduce this dependence by introducing auxiliary alignment or pseudo-label signals. TSPNet (Li et al., 2020) explored temporal-semantic correspondence, while GASLT (Yin et al., 2023) introduced gloss attention and transferred sentence-level similarity knowledge from a pretrained natural language model. ConSLT (Fu et al., 2023) introduced contextual cross-modal contrastive learning, and VAP (Jiao et al., 2024) employed visual alignment pretraining. Although these approaches reduce reliance on manual gloss supervision, they may still involve pseudo-glosses, pretrained recognition models, or additional pretraining stages.

2.2 Fully Gloss-Free SLT

Fully gloss-free SLT learns directly from paired sign videos and spoken-language translations without using intermediate gloss supervision. GFSLT-VLP (Zhou et al., 2023) introduced visual-language pretraining, while C^{2}RL (Chen et al., 2025) used cross-modal contrastive representation learning to improve semantic consistency between visual and textual representations. More recent systems, including FLa-LLM (Chen et al., 2024), SignLLM (Gong et al., 2024), and MMSLT (Kim et al., 2025), increasingly make use of large pretrained language or multimodal models to strengthen translation generation.

TransAlign-SLT takes a different approach by deriving complementary supervision directly from the paired translation signal. TranslationCTC provides a coarse temporal prior, while masked optimal transport supplies training-time semantic correspondence with contextualized word representations. This design keeps the inference pipeline fully gloss-free and avoids both pseudo-gloss generation and the use of a large pretrained language decoder.

3 Methodology

3.1 Method Overview

Given a continuous sign video $\mathbf{X} = \{x_1, \ldots, x_T\}$ and its corresponding spoken-language translation $\mathbf{Y} = \{y_1, \ldots, y_N\}$, fully gloss-free SLT learns a mapping $f_\theta : \mathbf{X} \rightarrow \mathbf{Y}$ without relying on intermediate gloss or pseudo-gloss supervision.

As shown in Figure 1, TransAlign-SLT combines a visual-temporal encoding pipeline with complementary temporal and semantic training signals. The encoding process is defined as

$$\mathbf{X} \xrightarrow{\phi_{\mathrm{CNN}}} \mathbf{F} \xrightarrow{\phi_{\mathrm{HMTF}}} \mathbf{H}^{\mathrm{conv}} \xrightarrow{\phi_{\mathrm{BiLSTM}}} \mathbf{H}^{\mathrm{seq}}, \quad (1)$$

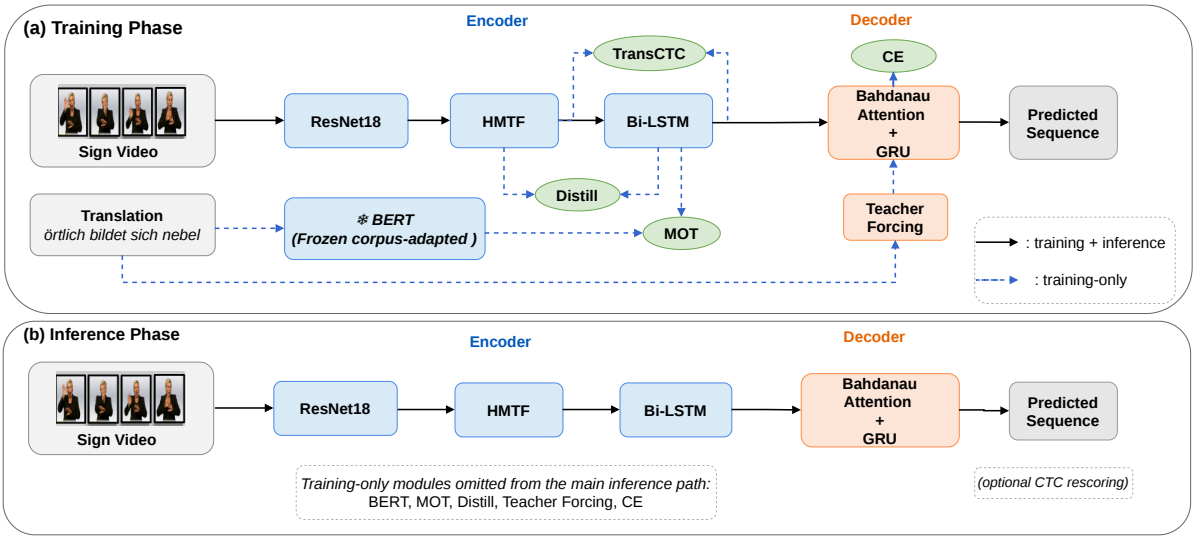


Figure 1: Overview of TransAlign-SLT during training and inference. ResNet18 features are processed by HMTF and BiLSTM before autoregressive translation with a Bahdanau-attention GRU. During training, TranslationCTC provides auxiliary temporal guidance, while frozen corpus-adapted BERT representations provide semantic supervision through MOT. BERT, MOT, distillation, teacher forcing, and training losses are excluded from the main inference path. TranslationCTC scores may optionally be used for post-hoc beam rescoring.

where ϕ_{CNN} is an ImageNet-pretrained ResNet18 (He et al., 2016; Deng et al., 2009), ϕ_{HMTF} denotes the Hierarchical Multi-scale Temporal Fusion encoder adapted from GPGN (Guo et al., 2024), and ϕ_{BiLSTM} provides bidirectional sequence refinement (Graves and Schmidhuber, 2005).

TranslationCTC encourages coarse temporal organization of the visual representations, while Masked Entropic Optimal Transport (MOT) provides soft visual-text semantic correspondence. Together, these signals improve representation learning from sentence-level translation supervision before an attention-based GRU generates the target sentence autoregressively.

3.2 Hierarchical Multi-scale Temporal Fusion

HMTF adapts the Parallel Densely-Connected Temporal Feature Extraction design of GPGN (Guo et al., 2024) to the translation setting. We adopt only the architectural design from GPGN and do not initialize HMTF using GPGN checkpoints or any gloss-supervised pretrained weights. Parallel dilated temporal convolutions capture information over multiple receptive fields, while cross-scale residual connections and hierarchical fusion integrate the resulting temporal representations. A detailed view of the HMTF architecture is shown in Figure 2.

Given frame-level features $\mathbf{F}$ and valid lengths $\mathbf{L}$, HMTF produces temporal representations, TranslationCTC logits, and encoded sequence lengths:

$$(\mathbf{H}^{\text{conv}}, \mathbf{Z}^{\text{conv}}, \boldsymbol{\ell}) = \phi_{\text{HMTF}}(\mathbf{F}, \mathbf{L}). \tag{2}$$

Here, $\mathbf{H}^{\text{conv}} \in \mathbb{R}^{\ell_{\max} \times B \times d_h}$ and $\mathbf{Z}^{\text{conv}} \in \mathbb{R}^{\ell_{\max} \times B \times V_{\text{ctc}}}$.

The fused temporal representations are then refined by a stacked BiLSTM:

$$\mathbf{H}^{\text{seq}} = \phi_{\text{BiLSTM}}(\mathbf{H}^{\text{conv}}, \boldsymbol{\ell}), \tag{3}$$

which introduces bidirectional sequence context that complements the multi-scale convolutional modeling performed by HMTF.

3.3 Auxiliary TranslationCTC Supervision

Without gloss supervision, there is no explicit intermediate correspondence between the visual sequence and linguistic units. We therefore apply TranslationCTC, based on Connectionist Temporal Classification (CTC) (Graves et al., 2006), to both the HMTF and BiLSTM representations as an auxiliary temporal regularizer.

Let $r \in \{\text{conv}, \text{seq}\}$ denote the two branches, and let $\mathbf{Z}_b^r$ denote their logits produced by a shared classifier. For sample b, the branch-specific objective is

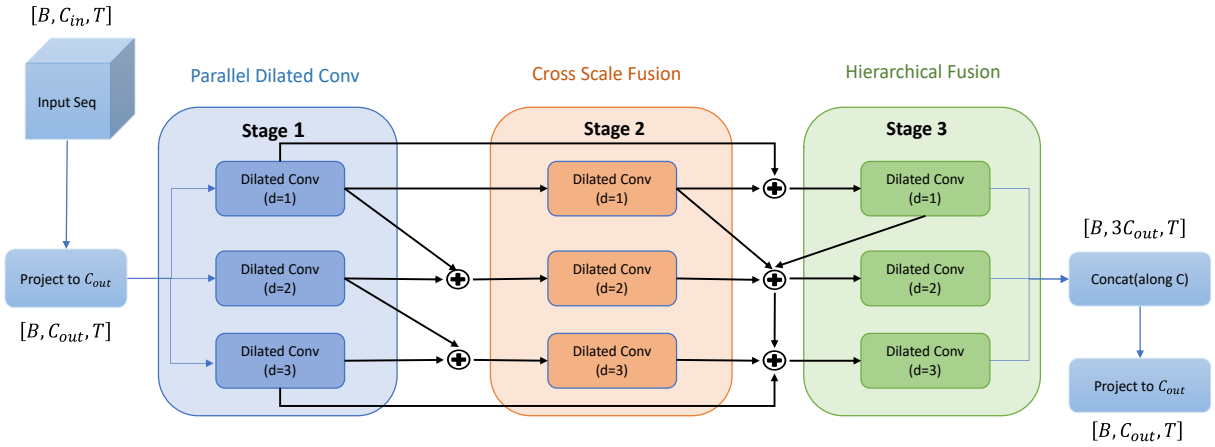


Figure 2: Hierarchical Multi-scale Temporal Fusion (HMTF). Parallel dilated convolutions capture multiple temporal scales, cross-scale residual connections exchange information across branches, and hierarchical fusion produces the final temporal representation.

$$\mathcal{L}_{\mathrm{CTC}}^{r} = \frac{1}{B}\sum_{b=1}^{B} \mathrm{CTC}\Big(\log \mathrm{softmax}(\mathbf{Z}_b^r), \mathbf{y}_b^{\mathrm{ctc}}, \ell_b, N_b\Big). \tag{4}$$

where $\mathbf{y}_b^{\mathrm{ctc}}$ is the target spoken-language token sequence, ℓ_b is the valid visual sequence length, and N_b denotes the corresponding target sequence length.

TranslationCTC uses the same translation vocabulary as the decoder, excluding PAD, BOS, and EOS tokens, with an additional CTC blank class. We use PyTorch CTCLoss with `zero_infinity=True` to handle invalid alignments.

During the early stages of training, the visual representations are only weakly organized with respect to the target language. As a result, fine-grained visual-semantic correspondence is difficult to learn directly from sentence-level supervision. TranslationCTC is therefore assigned a relatively strong initial weight, providing a coarse token-level temporal-linguistic prior that helps organize the visual sequence.

This weight is smoothly reduced from 0.5 to 0.1 over the course of training. Reducing its influence relaxes the monotonic auxiliary constraint and increases the relative contribution of MOT-based semantic alignment and autoregressive translation learning. This is important because sign-to-spoken-language translation does not necessarily follow a strictly monotonic frame-to-word correspondence. TranslationCTC therefore acts primarily as an early optimization prior rather than as the final alignment mechanism.

We further transfer sequence-level information from the BiLSTM branch to the convolutional branch through knowledge distillation:

$$\mathcal{L}_{\mathrm{Dist}} = \frac{\tau^2}{N_{\mathrm{valid}}} \sum_{b=1}^{B}\sum_{t=1}^{\ell_b} D_{\mathrm{KL}}\left(\tilde{\mathbf{p}}_{t,b}^{\mathrm{seq}} \,\|\, \tilde{\mathbf{p}}_{t,b}^{\mathrm{conv}}\right), \tag{5}$$

where τ is the distillation temperature and $N_{\mathrm{valid}} = \sum_b \ell_b$ is the total number of valid temporal positions. We set $\tau = 8$ in all experiments.

3.4 Masked Semantic Alignment with Frozen BERT

TranslationCTC provides temporal guidance, but it does not explicitly capture contextual sentence semantics. To complement this signal, we use **Masked Entropic Optimal Transport (MOT)** to establish soft semantic correspondence between visual temporal features and contextualized textual representations.

A language-specific BERT encoder (Devlin et al., 2019) is first adapted using only the training-set translations of the target corpus through masked language modeling and is then frozen during SLT training. Since WordPiece tokenization may divide a single lexical unit into multiple subwords, subwords associated with the same original word are mean-pooled before alignment, yielding $\mathbf{T}_b = \{\mathbf{t}_{b,j}\}_{j=1}^{T_b}$.

Each visual sequence feature is projected into the textual embedding space:

$$\mathbf{q}_{b,i} = \mathbf{W}_p \mathbf{h}_{b,i}^{\mathrm{seq}} + \mathbf{b}_p, \tag{6}$$

and the corresponding visual-text transport cost is defined using cosine distance:

$$\mathbf{C}_{ij}^{(b)} = 1 - \hat{\mathbf{q}}_{b,i}^{\top} \hat{\mathbf{t}}_{b,j}. \tag{7}$$

To prevent padded positions from contributing to the alignment, uniform marginals are normalized only over valid visual and textual positions:

$$\mu_{b,i} = \frac{m_{b,i}^{v}}{\sum_k m_{b,k}^{v}}, \qquad \nu_{b,j} = \frac{m_{b,j}^{t}}{\sum_k m_{b,k}^{t}}. \tag{8}$$

The entropically regularized transport plan is defined as

$$\boldsymbol{\pi}^{(b)} = \arg \min_{\mathbf{T} \in \Pi(\boldsymbol{\mu}_b, \boldsymbol{\nu}_b)} \left[\langle \mathbf{T}, \mathbf{C}^{(b)} \rangle - \epsilon \mathcal{H}(\mathbf{T}) \right], \tag{9}$$

with the semantic alignment objective

$$\mathcal{L}_{\mathrm{MOT}} = \frac{1}{B} \sum_{b=1}^{B} \left\langle \boldsymbol{\pi}^{(b)}, \mathbf{C}^{(b)} \right\rangle. \tag{10}$$

We solve Eq. 9 using 50 log-space Sinkhorn iterations (Cuturi, 2013) with $\epsilon = 0.1$. In contrast to TranslationCTC, MOT does not explicitly enforce monotonic frame-to-word correspondence. It therefore provides a complementary form of semantic supervision. The frozen BERT/MOT branch is used only during training.

3.5 Translation Decoder and CTC Rescoring

The target translation is generated autoregressively using a single-layer GRU with Bahdanau additive attention (Bahdanau et al., 2015; Camgoz et al., 2018), conditioned on $\mathbf{H}^{\mathrm{seq}}$. At decoding step u, the model predicts

$$P(y_u \mid y_{<u}, \mathbf{H}^{\mathrm{seq}}) = \mathrm{softmax}\Big(\mathbf{W}_o [\mathbf{s}_u; \mathbf{c}_u; \mathbf{x}_u] + \mathbf{b}_o \Big). \tag{11}$$

where $\mathbf{s}_u$ is the GRU hidden state and $\mathbf{c}_u$ denotes the Bahdanau attention context.

At inference, hypotheses are generated autoregressively using beam search and may optionally be reranked with TranslationCTC scores:

$$S_{\mathrm{final}} = (1 - \beta_{\mathrm{ctc}}) S_{\mathrm{beam}} + \beta_{\mathrm{ctc}} S_{\mathrm{CTC}}. \tag{12}$$

The autoregressive beam score is normalized using a length penalty, while the TranslationCTC score is normalized by the target hypothesis length before interpolation. The CTC score is applied only to completed hypotheses and does not participate in autoregressive beam expansion. Thus, it does not constrain the decoder's generation path, although its sequence score retains the monotonic alignment assumption of CTC.

3.6 Overall Training Objective

The complete training objective is

$$\mathcal{L}_{\mathrm{total}} = \lambda_{\mathrm{ctc}}(e) \left(\mathcal{L}_{\mathrm{CTC}}^{\mathrm{conv}} + \mathcal{L}_{\mathrm{CTC}}^{\mathrm{seq}} \right) + 10\, \mathcal{L}_{\mathrm{Dist}} + 2\, \mathcal{L}_{\mathrm{MOT}} + \mathcal{L}_{\mathrm{TransCE}}, \tag{13}$$

where $\lambda_{\mathrm{ctc}}(e)$ is the epoch-dependent TranslationCTC coefficient, smoothly reduced from 0.5 to 0.1. TranslationCTC provides coarse temporal regularization, MOT contributes contextual semantic alignment, and translation cross-entropy directly optimizes the final autoregressive translation objective.

4 Experiments

4.1 Datasets and Evaluation

Datasets. We evaluate TransAlign-SLT on RWTH-PHOENIX-Weather 2014T (PHOENIX14T) (Camgoz et al., 2018) and CSL-Daily (Zhou et al., 2021). PHOENIX14T consists of 7,096 training, 519 development, and 642 test video-translation pairs collected from German weather broadcasts. Its spoken-language vocabulary contains 2,887 words. CSL-Daily contains 18,401 training, 1,077 development, and 1,176 test examples covering daily-life communication, with a spoken-language vocabulary of 2,343 Chinese words. Although gloss annotations are available for both datasets, TransAlign-SLT uses only sign videos and spoken-language translations during training and inference.

Metrics.

Following standard SLT evaluation protocols, we report BLEU-1 through BLEU-4 (Papineni et al., 2002) together with ROUGE-L (Lin, 2004). BLEU scores are computed using NLTK's corpus-level BLEU implementation with smoothing method 3. ROUGE-L is computed as the mean sentence-level F1 score based on the longest common subsequence of token sequences. The translation text is lowercased, and punctuation is removed before dictionary encoding.

4.2 Implementation Details

We use a ResNet18 backbone initialized with ImageNet-pretrained weights from torchvision.

Following TSN-style sampling (Wang et al., 2016), 50% of the frames are retained during training by partitioning each video into uniformly spaced temporal segments and randomly sampling one frame from each segment. Each sampled frame is resized to 256×256, randomly cropped to 224×224, and horizontally flipped. During inference, we sample the same number of frames at uniformly spaced temporal positions and apply center cropping.

The semantic branch uses a language-specific BERT encoder with a hidden dimension of 768. The encoder is adapted using only the training-set translations of the corresponding benchmark, with MLM masking ratios of 0.5 for PHOENIX14T and 0.3 for CSL-Daily. After this adaptation, BERT remains frozen throughout SLT training and is not used during inference.

All models are trained for 60 epochs using Adam. For PHOENIX14T, both the learning rate and weight decay are set to 10^{-4}. For CSL-Daily, the learning rate is 5×10^{-5} and the weight decay is 10^{-6}. The TranslationCTC coefficient is gradually reduced from 0.5 to 0.1. MOT is computed using 50 log-space Sinkhorn iterations with $\epsilon = 0.1$. All experiments were conducted on an NVIDIA RTX A6000 GPU.

At inference, we use beam search with a beam size of 10, followed by post-hoc TranslationCTC reranking with $\beta_{\text{ctc}} = 0.6$. CTC scores are applied only to completed hypotheses and therefore do not affect autoregressive beam expansion. The deployed model contains 45.6M parameters and requires 123 GFLOPs for 64 RGB frames at 224×224. The frozen BERT/MOT branch is used only during training.

4.3 Main Results

Tables 1 and 2 summarize performance on both the development and test sets. All component-level ablations are conducted on the PHOENIX14T development split.

PHOENIX14T. TransAlign-SLT achieves 22.86 BLEU-4 / 47.63 ROUGE-L on the development set and 23.54 BLEU-4 / 48.46 ROUGE-L on the test set (Table 1). Among fully gloss-free systems, it outperforms GFSLT-VLP, FLa-LLM, and SignLLM in test BLEU-4, while STN-SLT, C^2RL, and MMSLT achieve higher scores. These results show that competitive translation performance can be obtained directly from video-translation pairs without gloss or pseudo-gloss supervision and without relying on a large pretrained language decoder at inference.

CSL-Daily. On CSL-Daily, TransAlign-SLT obtains 16.17 BLEU-4 / 44.13 ROUGE-L on the development set and 16.39 BLEU-4 / 44.68 ROUGE-L on the test set (Table 2). It again outperforms GFSLT-VLP, FLa-LLM, and SignLLM in test BLEU-4, while C^2RL and MMSLT achieve higher scores. These results show that TransAlign-SLT remains competitive across two distinct SLT benchmarks, while component-level ablations are conducted on PHOENIX14T.

4.4 Qualitative Analysis

Table 3 presents representative examples of successful semantic preservation together with remaining translation errors. In the first example, TransAlign-SLT preserves most of the sentence-level content while producing a slightly shorter translation. In the second example, the model retains the geographic and weather context but fails to preserve the distinction between *Schnee* (snow) and *Schneeregen* (sleet). The third example illustrates a fine-grained numerical error, where *fünften* (fifth) is replaced by *zehnten* (tenth). Taken together, these examples suggest that the model captures broad semantic content more reliably than fine-grained lexical and numerical details.

5 Ablation and Analysis

We study the contribution of temporal modeling, semantic supervision, TranslationCTC, decoding strategy, and decoder choice on the PHOENIX14T development set.

5.1 Temporal Modeling Components

We begin by examining the temporal modeling components of TransAlign-SLT. Table 4 compares the full HMTF + BiLSTM encoder with two controlled architectural variants.

Replacing HMTF with a conventional temporal convolution module while retaining the BiLSTM lowers BLEU-4 from 22.86 to 18.18, corresponding to a drop of 4.68 points. Keeping HMTF but removing the BiLSTM reduces BLEU-4 to 20.14 and ROUGE-L to 43.77. These results indicate that the two stages play complementary roles. HMTF captures temporal patterns at multiple scales, while the BiLSTM further refines these representations using bidirectional sequence context.

Method	Dev					Test				
	B1	B2	B3	B4	R	B1	B2	B3	B4	R
Gloss-supervised SLT										
SLRT (Camgoz et al., 2020)	47.26	34.40	27.05	22.38	–	46.61	33.73	26.19	21.32	–
STMC-T (Zhou et al., 2022)	47.60	36.43	29.18	24.09	48.24	46.98	36.09	28.70	23.65	46.65
BN-TIN-Transf.+SignBT (Zhou et al., 2021)	51.11	37.90	29.80	24.45	50.29	50.80	37.75	29.72	24.32	49.54
MMTLB (Chen et al., 2022a)	53.95	41.12	33.14	27.61	53.10	53.97	41.75	33.84	28.39	52.65
TS-SLT (Chen et al., 2022b)	54.32	41.99	34.15	28.66	54.08	54.90	42.43	34.46	28.95	53.48
SLTUNET (Zhang et al., 2023)	–	–	–	27.87	52.23	52.92	41.76	33.99	28.47	52.11
Weakly Supervised Gloss-free SLT										
TSPNet (Li et al., 2020)	–	–	–	–	–	36.10	23.12	16.88	13.41	34.96
GASLT (Yin et al., 2023)	–	–	–	–	–	39.07	26.74	21.86	15.74	39.86
ConSLT (Fu et al., 2023)	–	–	–	21.11	47.74	–	–	–	21.59	47.69
Sign2GPT (Wong et al., 2024)	–	–	–	–	–	49.54	35.96	28.83	22.52	48.90
VAP (Jiao et al., 2024)	52.78	–	–	26.62	51.47	53.07	–	–	26.16	51.28
Fully Gloss-free SLT										
NSLT (Camgoz et al., 2018)	28.10	16.81	11.82	9.12	31.00	27.10	15.61	10.82	8.35	29.70
NSLT+Bahdanau (Camgoz et al., 2018)	31.87	19.11	13.16	9.94	31.80	32.24	19.03	12.83	9.58	31.80
NSLT+Luong (Camgoz et al., 2018)	31.58	18.98	13.22	10.00	32.60	29.86	17.52	11.96	9.00	30.70
STN-SLT (Voskou et al., 2021)	**49.12**	36.29	28.34	23.23	–	48.61	35.97	28.37	23.65	–
CSGCR (Zhao et al., 2022)	35.85	24.77	18.65	15.08	38.96	36.71	25.40	18.86	15.18	38.85
GFSLT-VLP (Zhou et al., 2023)	44.08	33.56	26.74	22.12	43.72	43.71	33.18	26.11	21.44	42.49
FLa-LLM (Chen et al., 2024)	–	–	–	–	–	46.29	35.33	28.03	23.09	45.27
SignLLM (Gong et al., 2024)	46.88	<u>36.59</u>	<u>29.91</u>	<u>25.25</u>	47.23	45.21	34.78	28.05	23.40	44.49
C^2RL (Chen et al., 2025)	–	–	–	–	–	**52.81**	**40.20**	**32.20**	**26.75**	**50.96**
MMSLT (Kim et al., 2025)	<u>48.73</u>	**37.78**	**30.51**	**25.47**	**48.58**	<u>48.92</u>	<u>38.12</u>	<u>30.79</u>	<u>25.73</u>	47.97
TransAlign-SLT (Ours)	48.13	36.04	28.55	22.86	<u>47.63</u>	48.53	36.97	29.39	23.54	<u>48.46</u>

Table 1: Comparison with prior methods on PHOENIX14T. Methods are grouped by supervision setting. B1 to B4 denote BLEU-1 to BLEU-4, and R denotes ROUGE-L. Best and second-best results within the fully gloss-free setting are shown in bold and underlined, respectively.

Method	Dev					Test				
	B1	B2	B3	B4	R	B1	B2	B3	B4	R
Gloss-supervised SLT										
SLRT (Camgoz et al., 2020)	37.47	24.67	16.86	11.88	37.96	37.38	24.36	16.55	11.79	36.74
BN-TIN-Transf.+SignBT (Zhou et al., 2021)	51.46	37.23	27.51	20.80	49.49	51.42	37.26	27.76	21.34	49.31
MMTLB (Chen et al., 2022a)	53.81	40.84	31.29	24.42	53.38	53.31	40.41	30.87	23.92	53.25
Twostream-SLT (Chen et al., 2022b)	55.21	42.31	32.71	25.76	55.10	55.44	42.59	32.87	25.79	55.72
SLTUNET (Zhang et al., 2023)	–	–	–	23.99	53.58	54.98	41.44	31.84	25.01	54.08
Weakly Supervised Gloss-free SLT										
TSPNet* (Li et al., 2020)	–	–	–	–	–	17.09	8.98	5.07	2.97	18.38
GASLT (Yin et al., 2023)	–	–	–	–	–	19.90	9.94	5.98	4.07	20.35
ConSLT (Fu et al., 2023)	–	–	–	14.80	41.46	–	–	–	14.53	40.98
Sign2GPT (Wong et al., 2024)	–	–	–	–	–	41.75	28.73	20.60	15.40	42.36
VAP$^\diamond$ (Jiao et al., 2024)	50.41	–	–	21.16	48.72	49.99	–	–	20.85	48.56
Fully Gloss-free SLT										
SLRT-GF‡ (Camgoz et al., 2020; Zhou et al., 2023)	21.03	9.97	5.96	4.04	20.51	20.00	9.11	4.93	3.03	19.67
NSLT+Luong (Camgoz et al., 2018)	34.22	19.72	12.24	7.96	34.28	34.16	19.57	11.84	7.56	34.54
GFSLT-VLP (Zhou et al., 2023)	39.20	25.02	16.35	11.07	36.70	39.37	24.93	16.26	11.00	36.44
FLa-LLM (Chen et al., 2024)	–	–	–	–	–	37.13	25.12	18.38	14.20	37.25
SignLLM (Gong et al., 2024)	42.45	26.88	17.90	12.23	39.18	39.55	28.13	20.07	15.75	39.91
C^2RL (Chen et al., 2025)	–	–	–	–	–	<u>49.32</u>	<u>36.28</u>	**27.54**	**21.61**	<u>48.21</u>
MMSLT (Kim et al., 2025)	**50.05**	**36.39**	**26.91**	**20.51**	**48.53**	**49.87**	**36.37**	<u>27.29</u>	<u>21.11</u>	**48.92**
TransAlign-SLT (Ours)	<u>44.59</u>	<u>32.22</u>	<u>22.75</u>	<u>16.17</u>	<u>44.13</u>	44.59	32.21	22.89	16.39	44.68

Table 2: Comparison with prior methods on CSL-Daily (Zhou et al., 2021). Methods are grouped by supervision setting. B1 to B4 denote BLEU-1 to BLEU-4, and R denotes ROUGE-L. * denotes TSPNet results reproduced by prior work (Yin et al., 2023). ‡ denotes the SLRT architecture reproduced under the gloss-free setting by GFSLT-VLP (Zhou et al., 2023). For VAP (Jiao et al., 2024), marked with $^\diamond$, we report results without punctuation post-processing following (Min and Chen, 2023). Best and second-best results within the fully gloss-free setting are shown in bold and underlined, respectively.

Reference:	auch morgen erwartet uns eine ruhige herbstmischung aus hochnebel wolken und sonne (*tomorrow too we can expect a calm autumnal mix of high fog, clouds and sunshine*)
TransAlign-SLT:	morgen erwartet uns eine mischung aus hochnebel wolken und sonne (*tomorrow we can expect a mix of high fog, clouds and sunshine*)
Reference:	im osten und südosten auch schnee oder schneeregen (*in the east and southeast also snow or sleet*)
TransAlign-SLT:	im osten und südosten schnee oder schnee (*in the east and southeast snow or snow*)
Reference:	und nun die wettervorhersage für morgen samstag den fünften dezember (*and now the weather forecast for tomorrow, Saturday, 5 December*)
TransAlign-SLT:	und nun die wettervorhersage für morgen samstag den zehnten dezember (*and now the weather forecast for tomorrow, Saturday, 10 December*)

Table 3: Qualitative translations on the PHOENIX14T test set. Correctly preserved content is shown in blue, semantically related but inaccurate content is shown in green, and incorrect content is shown in red.

Temporal Encoder	**B4**	**R**
TemporalConv + BiLSTM	18.18	43.09
HMTF w/o BiLSTM	20.14	43.77
HMTF + BiLSTM	**22.86**	**47.63**

Table 4: Effect of temporal modeling components on PHOENIX14T Dev. B4 denotes BLEU-4 and R denotes ROUGE-L.

5.2 Semantic Supervision

Table 5 examines the individual components of the training-time semantic branch. The baseline removes BERT-based semantic guidance while keeping the remaining TransAlign-SLT components unchanged. Introducing frozen BERT representations through MOT increases BLEU-4 from 21.54 to 21.85 and ROUGE-L from 45.10 to 45.95.

Adapting BERT to the target translation corpus using masked language modeling provides a further improvement, reaching 22.54 BLEU-4 and 46.90 ROUGE-L. The best performance is obtained when WordPiece subwords are additionally pooled into word-level representations before transport alignment, yielding 22.86 BLEU-4 and 47.63 ROUGE-L. This result suggests that the semantic signal is more useful when the text representations are adapted to the target corpus and aligned at the word level rather than at fragmented subword positions.

Method	**MLM Adapt**	**Word Pool**	**MOT Align**	**B1**	**B2**	**B3**	**B4**	**R**
Baseline	✗	✗	✗	47.43	34.00	27.21	21.54	45.10
+ BERT	✗	✗	✓	47.50	34.23	27.88	21.85	45.95
+ MLM adaptation	✓	✗	✓	48.00	35.88	28.10	22.54	46.90
+ Word pooling	✓	✓	✓	**48.13**	**36.04**	**28.55**	**22.86**	**47.63**

Table 5: Effect of training-time semantic supervision on PHOENIX14T Dev. B1 to B4 denote BLEU-1 to BLEU-4, and R denotes ROUGE-L.

5.3 Effect of TranslationCTC

Table 6 studies both the strength and the scheduling of TranslationCTC supervision. Removing TranslationCTC ($\lambda = 0$) causes BLEU-4 to fall from 22.86 to 10.50 and ROUGE-L from 47.63 to 27.01, showing that this auxiliary signal is important during representation learning. At the same time, maintaining a strong constant weight is not optimal. A fixed weight of $\lambda = 0.6$ produces 21.50 BLEU-4, while moderate constant weights recover most of the performance.

The proposed $0.5 \rightarrow 0.1$ schedule achieves the best BLEU-2, BLEU-3, and BLEU-4 scores. This behavior supports the role of TranslationCTC as an early temporal-linguistic prior whose influence is gradually reduced as training progresses.

CTC Strategy	**B1**	**B2**	**B3**	**B4**	**R**
Constant ($\lambda = 0.0$)	16.58	13.56	11.20	10.50	27.01
Constant ($\lambda = 0.6$)	47.56	35.01	26.32	21.50	47.20
Constant ($\lambda = 0.5$)	48.32	35.93	28.24	22.42	47.46
Constant ($\lambda = 0.4$)	**48.50**	35.76	28.09	22.40	**47.70**
Decay ($0.5 \rightarrow 0.1$)	48.13	**36.04**	**28.55**	**22.86**	47.63

Table 6: Effect of TranslationCTC weighting on PHOENIX14T Dev.

Complementarity with MOT. TranslationCTC encourages coarse temporal organization, whereas MOT provides soft semantic correspondence without explicitly enforcing a monotonic frame-to-word alignment. The separate ablations show that both signals are useful. However, we do not interpret these results as a fully controlled factorial decomposition of their interaction.

5.4 Decoding with CTC Rescoring

Table 7 separates the contribution of autoregressive beam search from that of post-hoc TranslationCTC rescoring. Beam search accounts for most of the decoding improvement, raising BLEU-4 from 20.58 with greedy decoding to 22.40.

Applying TranslationCTC rescoring to completed beam hypotheses provides an additional improvement, reaching 22.86 BLEU-4 and 47.63 ROUGE-L. By comparison, CTC rescoring without beam search produces only a small gain over greedy decoding.

CTC scores are used only to rerank completed hypotheses and do not participate in beam expansion. TranslationCTC therefore does not impose a hard monotonic decoding path. Moreover, the relatively small gain from inference-time rescoring, compared with the large performance drop observed when TranslationCTC is removed during training, suggests that its main contribution comes from representation learning rather than from decoding-time reranking.

Decoding	Beam	CTC	B1	B2	B3	B4	R
Greedy	✗	✗	45.84	33.32	26.03	20.58	45.44
CTC rescore only	✗	✓	45.90	33.50	26.20	20.60	45.60
Beam	✓	✗	47.18	35.27	27.97	22.40	46.54
Beam + CTC	✓	✓	**48.13**	**36.04**	**28.55**	**22.86**	**47.63**

Table 7: Effect of beam search and post-hoc TranslationCTC rescoring on PHOENIX14T Dev.

5.5 Effect of Decoder Choice

We also examine whether the observed gains can be explained primarily by the choice of a larger pretrained language decoder. An mBART-based decoder variant achieves 20.43 BLEU-4 on PHOENIX14T Dev, compared with 22.86 BLEU-4 for the GRU-Bahdanau decoder used in TransAlign-SLT. This 2.43-point difference indicates that simply increasing decoder scale does not account for the improvements. Strong visual-temporal representations and training-time alignment remain important even when translation is generated with a lightweight recurrent decoder.

Translation quality versus model size. TransAlign-SLT achieves 23.54 test BLEU-4 with 45.6M inference parameters, compared with 25.73 BLEU-4 using 640.94M parameters for MMSLT (Kim et al., 2025) and 22.52 BLEU-4 using approximately 1.77B parameters for Sign2GPT (Wong et al., 2024). This indicates a favorable trade-off between translation quality and model size rather than establishing a direct runtime efficiency comparison. Figure 3 visualizes this trade-off on PHOENIX14T.

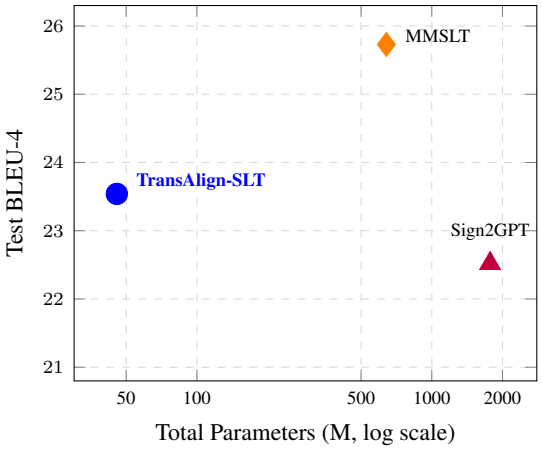


Figure 3: Translation quality versus model size on PHOENIX14T. The x-axis reports total model parameters on a logarithmic scale, and the y-axis reports test BLEU-4. Only methods with directly reported or measured complete model parameter counts are included.

Limitations and future work. Evaluation is limited to PHOENIX14T and CSL-Daily, so generalization beyond these benchmarks remains untested. We report model size and computational cost but not hardware-matched latency, memory, or training-time comparisons. Future work will consider broader sign languages and domains and controlled efficiency benchmarking.

6 Conclusion

We presented **TransAlign-SLT**, a fully gloss-free SLT framework that learns directly from paired sign videos and spoken-language translations using complementary temporal and semantic supervision. TranslationCTC provides coarse temporal regularization, while MOT encourages semantic alignment between visual features and contextualized word-level representations from a frozen, corpus-adapted BERT encoder. Experiments on PHOENIX14T and CSL-Daily, together with component-level ablations, show that these training signals improve learning from raw translation supervision while preserving a compact GRU-based inference pipeline that does not rely on glosses, pseudo-glosses, or a large pretrained language decoder. The results suggest a practical direction for scalable gloss-free SLT with compact inference pipelines.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations (ICLR)*.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7784–7793.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10023–10033.

Yutong Chen, Fangyun Wei, Xiao Sun, Zhirong Wu, and Stephen Lin. 2022a. A simple multi-modality transfer learning baseline for sign language translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5120–5130.

Yutong Chen, Ronglai Zuo, Fangyun Wei, Yu Wu, Shujie Liu, and Brian Mak. 2022b. Two-stream network for sign language recognition and translation. *Advances in Neural Information Processing Systems*, 35:17043–17056.

Zhigang Chen, Benjia Zhou, Yiqing Huang, Jun Wan, Yibo Hu, Hailin Shi, Yanyan Liang, Zhen Lei, and Du Zhang. 2025. C^2RL: Content and context representation learning for gloss-free sign language translation and retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(9):8533–8544.

Zhigang Chen, Benjia Zhou, Jun Li, Jun Wan, Zhen Lei, Ning Jiang, Quan Lu, and Guoqing Zhao. 2024. Factorized learning assisted with large language model for gloss-free sign language translation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7071–7081, Torino, Italia. ELRA and ICCL.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26, pages 2292–2300. Curran Associates, Inc.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Biao Fu, Peigen Ye, Liang Zhang, Pei Yu, Cong Hu, Xiaodong Shi, and Yidong Chen. 2023. A token-level contrastive framework for sign language translation. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Jia Gong, Lin Geng Foo, Yixuan He, Hossein Rahmani, and Jun Liu. 2024. LLMs are good sign language translators. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18362–18372.

Alex Graves, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber. 2006. Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks. In *International Conference on Machine Learning*, pages 369–376.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6):602–610.

Leming Guo, Wanli Xue, Bo Liu, Kaihua Zhang, Tiantian Yuan, and Dimitris Metaxas. 2024. Gloss prior guided visual feature learning for continuous sign language recognition. *IEEE Transactions on Image Processing*, 33:3486–3495.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Peiqi Jiao, Yuecong Min, and Xilin Chen. 2024. Visual Alignment Pre-Training for Sign Language Translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 349–367.

Jungeun Kim, Hyeongwoo Jeon, Jongseong Bae, and Ha Young Kim. 2025. Leveraging the power of MLLMs for gloss-free sign language translation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 21048–21058.

Dongxu Li, Chenchen Xu, Xin Yu, Kaihao Zhang, Ben Swift, Hanna Suominen, and Hongdong Li. 2020. TSPNet: Hierarchical feature learning via temporal semantic pyramid for sign language translation. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81.

Yuecong Min and Xilin Chen. 2023. Towards faithful sign language translation. Submitted to NeurIPS 2023.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Andreas Voskou, Konstantinos P. Panousis, Dimitrios Kosmopoulos, Dimitris N. Metaxas, and Sotirios Chatzis. 2021. Stochastic transformer networks with linear competing units: Application to end-to-end SL translation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11946–11955.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. 2016. Temporal segment networks: Towards good practices for deep action recognition. In *European Conference on Computer Vision (ECCV)*, pages 20–36.

Ryan Wong, Necati Cihan Camgoz, and Richard Bowden. 2024. Sign2GPT: Leveraging large language models for gloss-free sign language translation. In *International Conference on Learning Representations*, volume 2024, pages 18157–18174.

Aoxiong Yin, Tianyun Zhong, Li Tang, Weike Jin, Tao Jin, and Zhou Zhao. 2023. Gloss attention for gloss-free sign language translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2551–2562.

Biao Zhang, Mathias Müller, and Rico Sennrich. 2023. SLTUNET: A simple unified model for sign language translation. In *The Eleventh International Conference on Learning Representations*.

Jian Zhao, Weizhen Qi, Wengang Zhou, Nan Duan, Ming Zhou, and Houqiang Li. 2022. Conditional sentence generation and cross-modal reranking for sign language translation. *IEEE Transactions on Multimedia*, 24:2662–2672.

Benjia Zhou, Zhigang Chen, Albert Clapés, Jun Wan, Yanyan Liang, Sergio Escalera, Zhen Lei, and Du Zhang. 2023. Gloss-free sign language translation: Improving from visual-language pretraining. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20871–20881.

Hao Zhou, Wengang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. 2021. Improving sign language translation with monolingual data by sign back-translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1316–1325.

Hao Zhou, Wengang Zhou, Yun Zhou, and Houqiang Li. 2022. Spatial-temporal multi-cue network for sign language recognition and translation. *IEEE Transactions on Multimedia*, 24:768–779.

Articulator-Level Cross-Lingual Transfer in Sign Language Translation

Ali Elaswad[*]
The American University in Cairo
alyelaswad@aucegypt.edu

Omar Mokhtar Gaber Saqr[*]
The American University in Cairo
omar_saqr@aucegypt.edu

Abstract

Cross-lingual transfer is standard in low-resource sign language translation, but it is unclear what the transferred model provides. We fine-tune a pose-based model from American Sign Language (ASL) to Indian Sign Language (ISL) and decompose the transfer across its encoder, decoder, and articulator streams. The matched encoder-decoder pair reaches 3.61 ± 0.12 BLEU-4 over three seeds, while either component alone performs close to training from scratch. Content-word recall shows a sharp frequency effect: frequent content improves with training and data, whereas rare and unseen content stays near zero, even after twenty epochs and 49% more data. This is not a decoder artefact: decoded blind, the same checkpoint reproduces no frequency gradient, and frequency still orders recall within strata of contextual predictability. Comparing ASL and Chinese Sign Language sources with the output language held fixed, a 22% encoder advantage survives across seeds, and multi-seed weight analysis localises it to the hand, spatial, and body components; an apparent Chinese advantage on face and temporal does not survive a second seed. A second ASL corpus of a different domain reproduces the effect. Transfer supplies a coupled encoder-decoder starting point, but does not by itself solve the long tail of target-language content.

1 Introduction

Sign language translation has drawn increasing research interest, but progress remains concentrated on a small number of languages with large annotated corpora. For low-resource languages such as Indian Sign Language (ISL), a common strategy is cross-lingual transfer: pretrain on a higher-resource sign language and fine-tune on the target (Artiaga et al., 2024; Dal Bianco et al., 2025). Yet papers typically report only the resulting aggregate score, leaving open what the transferred model actually provides.

We study this question using a pose-based encoder-decoder model fine-tuned from American Sign Language (ASL) to ISL. We first decompose the transferred checkpoint by separating its sign encoder and language decoder. The matched pair reaches 3.61 ± 0.12 BLEU-4 over three seeds, while either component alone performs close to training from scratch. We then examine what the resulting model can actually translate. Content-word recall follows training frequency closely: frequent content improves with training and additional data, while rare and unseen content remains near zero even after twenty epochs and a 49% increase in training data. Two controls establish that this is a property of what the model reads rather than of its decoder: with the visual input removed the same checkpoint shows no frequency gradient, and frequency continues to order recall within strata of contextual predictability.

We additionally compare ASL and Chinese Sign Language (CSL) as source languages while holding the output language fixed. Most of the apparent gap between full checkpoints turns out to be the decoder's output language, leaving a 22% encoder advantage whose per-arm ranges do not overlap across seeds. Running the weight analysis over several seeds separates what survives from what does not: the ASL encoder starts measurably closer to ISL on the hand, spatial, and body components, while an apparent CSL advantage on the face and temporal components lies inside run-to-run spread. Because the ASL and CSL checkpoints differ in pretraining domain as well as language, we treat the surviving pattern as suggestive; a second ASL corpus of a different domain reproduces it, which

[*] These authors contributed equally to this work.

The iSign corpus (Joshi et al., 2024) and the Uni-Sign checkpoints (Li et al., 2025) we build on are public; Appendix A gives the source checkpoints, training configuration, per-condition initialisation, and evaluation protocol behind every number here.

Proceedings of the Second Workshop on Sign Language Processing, pages 224–236
October 29, 2026 ©2026 Association for Computational Linguistics

weakens a purely domain-based account without settling the question. Figure 1 further shows that the transferred model depends strongly on the body stream: removing it causes a larger degradation than removing either hand, while the hands alone provide almost no translation signal.

Our contributions are:

- We show that transfer from ASL to ISL requires the matched pretrained encoder-decoder pair; neither component alone recovers the transfer gain.
- We show that content-word recall is strongly tied to training frequency, with rare and unseen content remaining resistant to additional training and data.
- We isolate source-language effects from decoder output language and find an articulator-specific pattern, reported over multiple seeds so that the components which survive run-to-run variation are separated from those which do not, and whose interpretation remains limited by a source-domain confound.
- We audit our own evaluation: a corrected clip-to-video mapping exposing residual split leakage and its measured effect on every score, and two controls separating the vocabulary effect from the decoder's language model.

2 Related Work

Sign language translation has drawn growing attention as a way to bridge deaf and hearing communities (Camgoz et al., 2020; Joshi et al., 2024). Early systems relied on gloss annotation as an intermediate supervision signal, but glosses are costly to produce and unavailable for most sign languages, which has pushed recent work toward gloss-free translation. A common recipe pairs a visual encoder over the signing video, or its extracted pose keypoints, with a large pretrained language model as decoder (Zhou et al., 2023; Wong et al., 2024; Li et al., 2025), a design that trades some appearance detail for efficiency and signer invariance when keypoints replace raw frames (Dal Bianco et al., 2025).

Two properties of the languages bear on what follows. Sign languages mark interrogatives and negation non-manually, through brow position held across a clause rather than a lexical item (Zeshan, 2004), which makes the face stream a meaningful unit to ablate. Indian Sign Language is typologically independent of both American and Chinese Sign Language (Zeshan, 2003), so neither source is a close relative. The corpora also differ in domain — How2Sign is instructional monologue (Duarte et al., 2021), OpenASL interpreted news (Shi et al., 2022), CSL-Daily scripted conversation (Zhou et al., 2021) — a difference we use as a partial control in Section 4.4.

Because sign language data is scarce, a substantial line of work looks outside the target language for supervision. Scaling the pretraining corpus, model size, and number of source languages together improves translation broadly, though quality remains far from usable (Zhang et al., 2024). Multilingual pretraining followed by brief target-language fine-tuning reaches 2.71 BLEU-4 from a 2.61 monolingual baseline on ISLTranslate (Joshi et al., 2023), a different ISL corpus, and is the closest prior *transfer* result to ours (Dal Bianco et al., 2025). A complementary approach stitches poses for words drawn from recognition datasets into synthetic training pairs, raising iSign — the corpus we use — from 0.55 to 3.43 (Joshi et al., 2025); because it can only recombine signs it already has isolated examples for, it bounds what lexicon-internal augmentation can buy. Transfer between sign languages is established for isolated-sign recognition (Artiaga et al., 2024), but translation, with its encoder–decoder structure and open output space, raises the further question of which components carry the benefit. Pose is not the only route: video features augmented with optical flow reach 8.58 BLEU-4 on iSign (Chowdhury and Sanyal, 2025), well above any pose-only system including ours. We study what transfer contributes within a pose-only architecture, not what the corpus admits overall.

Two recent studies bear on our findings without addressing them directly. A self-supervised model over separate hand, face, and body streams reports a masking ablation on ASL consistent with ours: hands alone reach 0.2 BLEU-4, comparable to blind decoding, against 2.4 for all streams (Gueuwou et al., 2025). At a stronger operating point on British Sign Language, remaining failure modes include an inability to distinguish questions from statements, missed negations, and difficulty with fingerspelling (Jang et al., 2025) — all of which we observe independently.

Closest to our concerns about evaluation, signer-

dependent splits are shown to inflate reported scores, with BLEU-4 on one benchmark falling by more than half under signer-independent evaluation (Artiaga et al., 2025). We split by source video for this reason; Section 4 audits how well our implementation of that rule held and what its residual failure was worth. Signer identity is not distributed with the corpus, so signer non-overlap cannot be checked directly. Our benchmark anticipates a related concern, noting that names are fingerspelled and then assigned a short sign, and that clip segmentation often separates a name's introduction from its later use (Joshi et al., 2024).

3 Methodology

3.1 Model

We use Uni-Sign (Li et al., 2025) in its pose-only configuration. Four ST-GCN streams (Yan et al., 2018), one each for body, left hand, right hand, and face, read pose keypoints and produce pooled per-stream representations. These are concatenated and projected into an mT5-base decoder (Xue et al., 2021), which we refer to throughout as the sign encoder and the language decoder respectively. The encoder reads 69 of the 133 COCO-WholeBody keypoints (Jin et al., 2020): 9 body, 21 per hand, and 18 face. Uni-Sign ties the left- and right-hand modules, so the two hands share one set of weights; we report them jointly as *hands* throughout. The two halves are architecturally separable: either can be loaded from a pretrained checkpoint while the other is left at its own pretrained initialisation, or replaced entirely with a checkpoint pretrained on a different source language. This separability is what makes the decomposition and source-language experiments in Section 4 possible without any change to the architecture itself. Where we test the pretrained encoder in isolation, we keep it frozen; this probes the pretrained representation directly rather than testing whether a trainable encoder-only configuration would also underperform.

Source checkpoints. We do not pretrain: all three source checkpoints are the public pose-only translation releases accompanying Li et al. (2025), which share a common stage-1 pose pretraining on CSL-News and differ only in the translation corpus each was subsequently fine-tuned on. We use `how2sign`, `openasl`, and `csl_daily`, trained respectively on How2Sign (Duarte et al., 2021), OpenASL (Shi et al., 2022), and CSL-Daily (Zhou et al., 2021). Because stage 1 is shared, a comparison between two of these checkpoints isolates the stage-2 translation corpus, which is what varies in sign language and in domain; it does not isolate language from domain, and Section 4.4 returns to that. Appendix A lists file identifiers, parameter counts, and exactly which tensors each condition loads, freezes, and trains.

3.2 Isolating the Source Language

A full checkpoint pretrained on one sign language differs from a full checkpoint pretrained on another in two ways at once: the sign language the encoder was trained to read, and the spoken language the decoder was trained to write. Comparing two full checkpoints therefore conflates both effects, and cannot tell us which one is responsible for an observed gap.

We separate them by constructing a hybrid model: the sign encoder from one source-language checkpoint, paired with the language decoder from another. Concretely, we pair a Chinese Sign Language (CSL) encoder with the English-writing decoder from the ASL checkpoint. This hybrid isolates the encoder's source-language effect, since the decoder's output language is now held constant across every comparison. We separately confirmed that the ordering between source languages was unchanged when one arm was rerun under the other's training configuration, which rules out incidental training differences as the explanation for the ordering; we did not, however, control for the fact that the two source checkpoints' pretraining data differ in domain as well as language, which we return to in Section 4.4.

3.3 Locating the Advantage Across Articulators

Where a source-language advantage exists, we ask where in the signer's body it lives, using two methods that draw on different signals. The first measures how much each source encoder's weights change during ISL fine-tuning, separately for each of five components: hands, spatial configuration, body, face, and temporal movement. We define relative weight change for component c as

$$\Delta_c = \frac{\|\theta_c^{\text{fine-tuned}} - \theta_c^{\text{pretrained}}\|}{\|\theta_c^{\text{pretrained}}\|} \times 100\%, \qquad (1)$$

where $\theta_c^{\text{pretrained}}$ is the component's weights immediately after loading the source-language checkpoint and $\theta_c^{\text{fine-tuned}}$ is the same component's

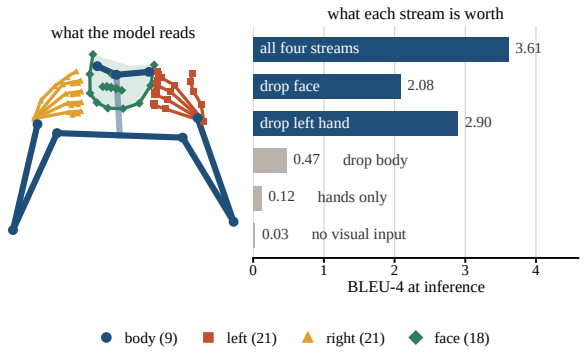


Figure 1: Left: keypoints supplied to each pose stream for a test clip. Right: BLEU-4 when streams are zeroed at inference. Removing the body stream causes the largest single-stream degradation, while the hands alone provide almost no translation signal.

weights after ISL fine-tuning completes. A smaller Δ_c means the source encoder's representation for component c needed less adjustment to fit ISL. Integer buffers such as BatchNorm step counters are excluded; squared in a norm, they would otherwise dominate every component.

The five components are *two overlapping partitions* of the same encoder, not five disjoint parts, and they do not sum. Hands, body, and face partition it by stream (74 tensors each); spatial and temporal partition it by layer type within the ST-GCN blocks (132 and 156 tensors), cutting across all four streams. The temporal component shows the largest relative change in every run, partly because its pretrained weights have the smallest norm, so an equal displacement registers as a larger ratio. Δ_c is therefore comparable between two sources *within* a component, not between components, and we treat it as directional rather than calibrated evidence: there is no control arm against which "small" is defined. We report it over two ASL and two CSL runs in one training configuration, and call a component separating only when the per-arm ranges do not overlap.

That second method measures behavioural recall directly: for content tied to a specific articulator, hand-carried proper nouns and numerals, or face-marked interrogatives, we measure how often the fine-tuned model reproduces that content correctly, with the decoder held fixed as described above. We define content-word recall as

$$R_{\text{word}} = \frac{1}{|D|} \sum_{d \in D} \frac{|\hat{w}_d \cap w_d|}{|w_d|}, \qquad (2)$$

where D is the evaluation set of clips, w_d is the set of target content words for clip d, and $\hat{w}_d$ is the set of content words the model actually produces for that clip. Question recall, reported in Sections 4.4 and 4.7, is a distinct quantity: the fraction of interrogative references rendered as questions in the output, and is not computed by Equation 2.

Both Δ_c and R_{word} are computed from the same underlying fine-tuning runs, so agreement between them is evidence that a pattern is not an artefact of one measurement method, but it is not evidence that the pattern is independent of confounds shared by both runs, including the pretraining domain confound noted above.

Finally, because the model's four pose streams are concatenated rather than fused earlier in the network, any subset of them can be zeroed at inference without retraining. We use this to run an articulator masking ablation directly on the trained model, which gives a third, purely behavioural view of which streams the model actually needs, independent of any source-language comparison.

4 Experiments

4.1 Experimental Setup

We use a verified-representative subsample of the iSign pose corpus, 23,921 of its 127,237 clips: per-decile counts range from 2332 to 2457 against an expected 2392, and mean reference length matches the corpus to within 0.04 words, 10.76 against 10.72. iSign defines no canonical split; we follow the dataset authors' recommendation to split by source video, obtaining 19,093 training, 2,396 development, and 2,432 test clips. Published results on this benchmark use self-constructed splits, so cross-paper comparisons throughout this section are indicative rather than exact.

Split audit. Our first clip-to-video rule was wrong, fragmenting single videos whose identifiers contain hyphens (Appendix B). Corrected, the splits hold 3,198, 411, and 413 source videos, and 22 videos cross the train–test boundary, contributing 0.90% of test clips. Removing them *raises* BLEU-4 by $+0.02$ to $+0.03$ on all three ASL seeds, since those clips are harder than average: the leakage is real but inflates nothing. Signer identity is not released with iSign, so signer-level independence cannot be checked directly.

Inputs are converted from MediaPipe Holistic (Lugaresi et al., 2019) to the COCO-WholeBody-133 layout the model expects (Jin et al., 2020). Every condition trains for 20 epochs with AdamW

(Loshchilov and Hutter, 2019) under a one-cycle schedule at peak learning rate 3×10^{-4}, effective batch 8, and bf16 autocast, on a single 16 GB or 24 GB consumer GPU. We take the final epoch rather than selecting on development score, so no condition benefits from checkpoint selection the others did not get; Appendix A gives the full configuration. Evaluation uses beam search with 4 beams and a 100-token limit, matching the reference implementation, and we report BLEU-4 (Papineni et al., 2002) and chrF (Popović, 2015) through `sacrebleu` (Post, 2018).

Seeds, and which set numbers come from. Every number below is a full test-set score, and conditions run at up to three seeds; we report the mean, the population standard deviation, and the individual runs. With so few runs, that spread describes the runs we have rather than a sampling distribution, so our claims turn on whether two arms' ranges overlap. Arms are compared only within a training configuration, since micro-batch size changes padding (Appendix A). Inference-time analyses needing one fixed model (Sections 4.5–4.6) use the seed-13 checkpoint, 3.58 BLEU-4 and 21.85 chrF standalone; bf16 decoding is not bit-deterministic, so it varies by up to ± 0.03 BLEU-4 between passes and each table carries its own control.

We validated the evaluation harness independently: the released ASL checkpoint scores 9.59 BLEU-4 and 30.28 chrF on How2Sign through our pipeline, matching its published figure, so the low absolute scores that follow reflect the difficulty of the ISL task rather than a fault in the harness.

4.2 Transfer Requires Both Components

Table 1 decomposes the transferred checkpoint on the test set. Every partial transfer we can construct, whether half the network or a single pose stream, lands between 0.31 and 0.76 BLEU-4, a band that straddles training from scratch rather than sitting above it, while the matched pair reaches 3.61 ± 0.12: roughly five times the best of them. Under this decomposition, the transferred benefit is not recovered by either component in isolation, but by the matched encoder-decoder pair. This has a direct practical consequence: replacing either half after transfer, for instance swapping in a stronger decoder, discards this relationship even when the replacement is individually stronger on its own.

One caveat: the frozen-encoder arm has by far the widest spread, 1.12 against 0.41 and 0.47, so its mean of 0.67 rests on one high run and is not a tier above the other partial transfers. What is robust is the gap between *any* partial transfer and the matched pair, an order of magnitude beyond the largest within-arm spread (Appendix A notes the configurations).

Transferred	**BLEU-4**	**Runs**
Encoder + decoder	$\mathbf{3.61 \pm 0.12}$	3.58, 3.77, 3.47
Encoder only (frozen)	0.67 ± 0.32	1.12, 0.41, 0.47
Decoder only	0.76	0.76
Hand streams only	0.61 ± 0.08	0.70, 0.53
Body stream only	0.58	0.58
Face stream only	0.31	0.31
Neither (scratch)	0.455 ± 0.001	0.454, 0.456

Table 1: Test-set decomposition of an ASL→ISL transfer. No partial transfer materially exceeds scratch; only the matched pair does. Spread is population standard deviation; the scratch row is given to three decimals because its two runs differ by 0.002. chrF in Appendix A. For reference, Joshi et al. (2025) report 3.43 BLEU-4 on iSign, the corpus we use, and Dal Bianco et al. (2025) 2.71 via multilingual transfer on ISLTranslate.

4.3 Source Language, Isolated from Output Language

Table 2 reports the intact CSL checkpoint alongside the hybrid configurations described in Section 3. Comparing full checkpoints within one configuration, ASL reaches 3.61 and CSL 0.59: an apparent gap of 3.02 BLEU-4, almost all of it the decoder's output language. With the decoder held fixed and both arms in the same configuration, the gap is 0.68 (3.75 against 3.07) — a 22% encoder advantage, not a fivefold one. That 22% is not a single-run artefact: the two arms' ranges, 3.67–3.82 against 3.06–3.09, do not overlap.

Both checkpoints share stage-1 pretraining and differ in the sign language *and* the domain of the stage-2 corpus. A partial control is available: OpenASL is also ASL but is interpreted news rather than instructional monologue, differing from How2Sign in domain about as much as CSL-Daily does. Its encoder reaches 3.54 with the same English decoder, close to How2Sign's 3.61 in that configuration and well clear of CSL's 3.07, while the configuration difference is worth only $+0.14$. Two ASL corpora of different domains behave alike and both beat CSL, which weakens a purely domain-based account without eliminating it: both remain more hand-forward than scripted conversation.

Encoder	Dec.	n	BLEU-4	chrF
matched configuration, same machine				
How2Sign (ASL)	En	2	$\mathbf{3.75 \pm 0.08}$	21.82
CSL-Daily (CSL)	En	2	3.07 ± 0.01	20.39
second configuration				
How2Sign (ASL)	En	3	3.61 ± 0.12	21.80
OpenASL (ASL)	En	1	3.54	21.82
CSL-Daily (CSL)	Zh	1	0.59	15.61

Table 2: Test scores separating source sign language from output language. Most of the naive ASL–CSL gap is the decoder; a 22% encoder advantage survives with the decoder held fixed, with non-overlapping per-arm ranges. OpenASL is a second ASL source of a different domain.

4.4 Where the Advantage Lives

We report this analysis as a suggestive pattern, for the reasons noted in Section 3. Table 3 and Figure 2 report weight change by component over two ASL and two CSL runs, all in one training configuration.

ASL starts closer to ISL on hands, spatial configuration, and body, with ranges that do not overlap. Face and temporal do not separate: face spans 415–492 across ASL runs against 365–515 across CSL, and temporal 264–293 against 281–326. A single run per source had shown CSL closer on both, which invited a typological reading of that split; it was an artefact of that run's ASL face value, 718.6, and we withdraw the claim. A third ASL run in the other configuration leaves which components separate unchanged.

Behavioural recall, computed as in Equation 2 with the decoder held fixed and paired at the item level, shows the same pattern: ASL leads on proper nouns (20.7% against 18.2%, 95% CI $[+1.4, +3.6]$) and numerals (35.9% against 29.3%, $[+3.9, +9.3]$), content carried mostly by hand shape, while face-marked questions do not separate (36.9% against 31.0% on 84 clips, interval $[-2.4, +14.9]$). Two measurements on different signals agree on which components resolve.

One explanation is that ASL is linguistically closer to ISL in hand configuration. An equally consistent explanation is that our ASL source is instructional monologue, which tends to be lexically dense and hand-forward, while our CSL source is scripted conversation. The OpenASL arm of Section 4.3 favours the first without settling it, and we flag this as the main open question raised by our source-language experiments rather than resolving it here.

Component	ASL (n=2)	CSL (n=2)	Closer
Hands	142.1 ± 1.6	157.9 ± 3.2	ASL
Spatial	50.5 ± 2.3	58.2 ± 1.9	ASL
Body	174.5 ± 7.8	217.1 ± 21.1	ASL
Face	453.9 ± 38.5	440.0 ± 74.9	—
Temporal	278.4 ± 14.3	303.3 ± 22.8	—

Table 3: Relative weight change during ISL fine-tuning (Δ_c, Equation 1), mean and population standard deviation over two runs per source in one configuration. Smaller means the source's pretrained representation needed less adjustment for ISL. "—" marks components whose per-arm ranges overlap, so the two sources do not separate. Hands, body, and face partition the encoder by stream; spatial (graph convolution over joints) and temporal (convolution over time) partition it by layer type, so the five rows are two overlapping views and do not sum.

Condition	BLEU-4	chrF	Date recall
Intact	3.59	21.80	65.8%
Shuffled poses	0.25	14.25	0.9%
Blind (all masked)	0.03	13.96	0.0%

Table 4: Grounding controls, test set, all decoded in one pass from the seed-13 checkpoint. Removing or scrambling the visual input destroys performance. The intact row is this pass's own control and reads 3.59 against the 3.58 of standalone evaluation and the 3.61 of the masking pass in Appendix E; the spread is decode nondeterminism, not a different model.

4.5 Grounding

Before relying on any of the above, we check that the model is reading the signing rather than generating plausible English independent of it. Blind decoding zeroes every pose stream at inference. Shuffled decoding feeds each clip a different clip's pose sequence while keeping its own reference, a stronger control since the input remains well-formed sign motion, just the wrong motion. We track date recall alongside BLEU, since dates cannot be guessed from a language prior and are exactly measurable.

Both controls collapse performance (Table 4): 0 of 111 dates survive blind decoding, and only 1 survives shuffling. Blind chrF, 13.96, falls below the length-matched random-sentence baseline of 15.14, meaning the model without visual input performs worse than random guessing calibrated to sentence length. The model's output depends on reading the visual input, and that blind decode is reused in Section 4.8 as the language-prior control.

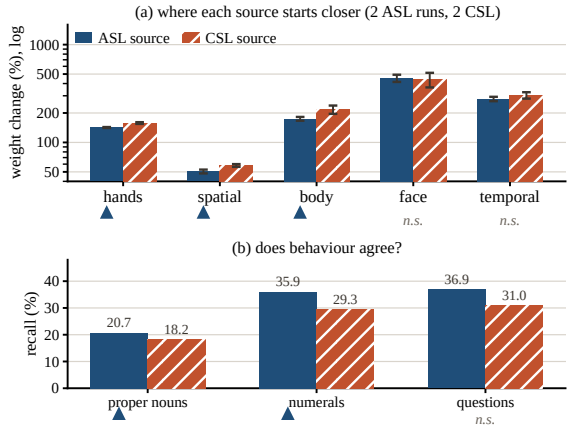


Figure 2: Where the ASL and CSL source encoders differ. (a) Relative weight change during ISL fine-tuning by component, log scale, two runs per source in one configuration, whiskers spanning their range. (b) Behavioural recall for content tied to each articulator, with the decoder held fixed. A triangle marks the source that wins; "n.s." marks a comparison whose ranges overlap (a) or whose bootstrap interval spans zero (b). The two panels agree: hand-carried measures separate, face-marked ones do not.

4.6 Articulator Masking

Because the four pose streams are concatenated rather than fused, any subset can be zeroed at inference without retraining. Figure 1 illustrates the four input streams and the resulting BLEU-4 changes, while Appendix E reports all sixteen stream combinations with chrF and date recall. Every configuration lacking the body stream collapses.

The body stream is necessary but not sufficient. Above size one, a subset containing body beats every same-sized subset without it, and removing body from the full set costs $3.61 \rightarrow 0.47$, more than removing any other stream. Yet body *alone* reaches 0.05, below hands alone at 0.12 and indistinguishable from blind decoding; no single stream exceeds 0.10. What the model needs is the conjunction: body with one hand reaches 1.26 and recovers 55% of dates, while both hands without body reach 0.12 and recover 0.9%. Hand configuration appears interpretable only relative to where the hands are, which the architecture encodes by positioning the hand streams on body keypoints. The hands are also not interchangeable: removing the right costs $3.61 \rightarrow 1.09$ against 2.90 for the left.

4.7 Non-Manual Marking

Sign languages mark interrogatives non-manually, principally through eyebrow position held across the clause (Zeshan, 2004). The face stream in this architecture samples jaw contour, inner mouth, and nose tip; mapped onto the standard 68-point face scheme (Sagonas et al., 2016), none of its 18 landmarks fall on the brows. Behaviour matches this gap: the seed-13 checkpoint renders 33.3% of interrogative references as questions, against 97.0% of statements kept as statements. (Section 4.4 quotes 36.9% for a different arm; 84 clips cannot resolve it.) Transfer lifts interrogative rendering (16.7% scratch, 25.0% frozen encoder, 33.3% matched pair) but it stays far behind the statement rate, and behind content grounding over the same three models, where outputs sharing no content word with the reference fall from 93.5% to 49.9%.

	BLEU-4		Q recall	
Seed	brow	jaw	brow	jaw
13	3.46	3.58	44.0%	33.3%
21	3.45	3.77	34.5%	39.3%
37	3.59	3.47	38.1%	42.9%
mean	3.50	3.61	38.9%	38.5%

Table 5: Eyebrow landmarks versus jaw contour, three seeds per arm. No measurable effect: both arms' ranges overlap on both measures.

We tested the implied fix directly: refilling the jaw-contour slots with eyebrow landmarks, keeping the stream size fixed so pretrained weights still load, three seeds per arm with matched jaw controls (Table 5). It does not measurably help. Pooled over seeds, question recall is 38.9% with eyebrows against 38.5% without, a difference of 0.4 points ($z = +0.09$, $p = 0.93$), with translation quality unchanged. A single seed would have suggested otherwise: seed 13 alone gives eyebrows a 10.7-point advantage, pairing that arm's best seed with the jaw arm's worst. Within-arm spread, 34.5–44.0% for brow and 33.3–42.9% for jaw, dwarfs the 0.4-point pooled difference. Because the pretrained face weights were fit to jaw-contour statistics, we cannot separate a capacity limit from a structural mismatch, making this an observation rather than a controlled test. Both readings predict the same lesson: adding non-manual features without adapting the capacity devoted to them is unlikely to help.

4.8 Vocabulary

This is, independent of the source-language comparisons above, our most consequential finding. Content-word recall, computed as in Equation 2, tracks closely with how often each reference word

Behaviour	Reference	Model output
Anchors survive, context invented	The Queen of the United Kingdom – Queen Elizabeth passed away on the afternoon of 8th September 2022.	However in a devastating incident, a beautiful girl named Queen Elizabeth died on 8th September 2022.
Unseen name dropped	DWWF was founded in 2014 and has been working ever since.	It was founded in 2014 and has been working ever since.
Correct, BLEU-penalised	On 3rd July Rahul Gandhi has made his resignation official and final.	On 3rd July, Rahul Gandhi announced his retirement.
Polarity reversed	This is the first time he is visiting the state since the removal of Article 370 in 2019.	This is not the first time such a case was reported in 2019 and was not the first time such a case was reported.
Degenerate repetition	On 15th June, 20 Indian soldiers were martyred in a violent face-off with Chinese troops in Galwan Valley.	On 15th June, the Chinese army participated in a fight against the Chinese army.

Table 6: Test-set outputs from the ASL-transferred model, one per behaviour. Dates and frequent names survive, unseen names vanish without a trace, and the surrounding text stays fluent whether or not it is accurate.

Predict.	unseen	rare	mid	freq.	v. freq.
low	0.000	0.002	0.072	0.143	0.173
mid	0.002	0.010	0.079	0.172	0.232
high	—	0.056	0.163	0.246	0.271

Table 7: Content-word recall by training frequency within terciles of contextual predictability. Frequency orders recall inside every tercile, so the frequency effect is not reducible to formulaic context. "—" marks a cell with too few tokens to estimate (142 or fewer elsewhere in that row).

Training frequency	n	Intact	Blind
Unseen (0)	1,251	0.001	0.000
Rare (1–4)	1,907	0.009	0.000
Mid (5–49)	5,920	0.096	0.002
Frequent (50–499)	6,397	0.203	0.016
Very frequent (500+)	1,320	0.248	0.001

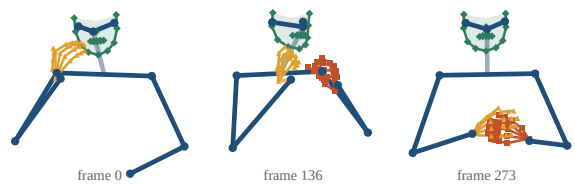
Table 8: Content-word recall by training frequency, with the same checkpoint decoded blind as a language-prior control. Words unseen in training are essentially never produced, and the blind column shows no frequency gradient: the effect belongs to what the model reads, not to its decoder.

appeared in training (Table 8), spanning more than two orders of magnitude between unseen and very frequent words.

Is this the decoder rather than the encoder? A frequent word is also likelier to sit in a formulaic context the decoder can pattern-match, so the curve could be a language-model effect. Two controls say it is not. Decoding the same checkpoint with every pose stream zeroed isolates the language prior, and that prior shows no frequency gradient: recall is ≤ 0.002 in four of five bands and peaks at 0.016, against 0.248 intact (Table 8, right column). Second, estimating each word's contextual predictability from its left context under a trigram model fitted to the training references, frequency still orders recall inside every tercile and the unseen band stays at the floor in all of them (Table 7). A logistic fit on both standardised predictors puts log frequency at $\beta = +0.73$ against log predictability at $\beta = +0.25$.

Restricting to proper nouns, which must be fingerspelled and so give a cleaner test of reading than ordinary vocabulary, the pattern sharpens: 0 of 370 proper-noun tokens unseen in training are ever reproduced, against 49.8% of those seen 50 times or more. We detect proper nouns orthographically, as capitalised tokens in non-sentence-initial position with possessives normalised, a rule with no dependence on a named-entity model; the same count run over the data-scale pair below gives 0 of 1,130 and 0 of 465. Figure 3 shows what this looks like in a single test clip: everything the model has seen in training survives, and the one word absent from training, a name, is replaced by a pronoun with no signal that anything is missing. Measured in aggregate, proper-noun recall of 0.216 looks reasonable; that aggregate is carried entirely by high-frequency names inferable from topic, and hides that recall on unseen names is exactly zero.

To interpret the headline score itself, we anchor it against trivial baselines that require no model at all (Appendix D). A random training sentence of matched length reaches chrF 15.14, close to what training from scratch achieves (15.29 over two seeds); the transferred model reaches 24 times the random baseline on BLEU-4, so the gain over these floors is real, even though the floors themselves sit



ref On 23rd June the Delhi High Court granted bail to Safoora.

out On 23rd June the Delhi High Court granted her bail.

Figure 3: A test clip, evenly sampled, with its reference and the model's output. Everything the model has seen in training survives: the date, the institution, the verb phrase. The one word absent from training, a name, is replaced by a pronoun, with nothing in the output marking that a name was there.

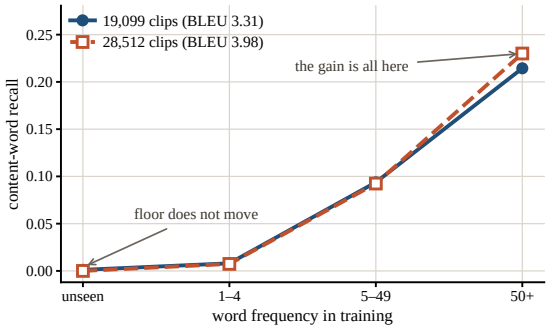


Figure 4: Content-word recall by training frequency for the controlled data-scale pair, evaluated on the same test set with frequency counted in each arm's own training corpus. The smaller arm is the mean of two seeds. Half again as much data lifts the frequent end of the curve and leaves the unseen end pinned at zero.

higher than an unanchored score would suggest.

The same pattern holds under scaling. A controlled comparison on a matched test set, one arm trained on 19,099 clips and the other on 28,512 (a separate corpus draw, Appendix A), shows recall rising at the frequent end while the unseen end stays at zero (Figure 4). Aggregate score does improve, from 3.31 ± 0.02 to 3.98 BLEU-4, a 21% relative gain from 49% more data, but recall on frequent words rises only from 0.214 to 0.230 and neither arm reproduces a single unseen proper noun. More training data behaves like more transfer: it improves the vocabulary the model already has some purchase on, not the vocabulary it has never seen. Our own headline result should be read in the same light. It uses 19,093 training clips, 15.0% of the 127,237-pair corpus (our full subsample is 18.8%). That makes the result data-efficient in aggregate score and not in any sense open-vocabulary, since both arms share the same floor.

We also checked whether pose estimation quality, rather than the model itself, explains this pattern. Stratifying test clips by hand-keypoint coverage rejects that explanation: the best coverage quartile reaches content recall of 0.152, against 0.108 in the worst, with correlation $r = +0.088$ across clips. Near-perfect hand tracking yields nearly identical failure, so improving keypoint extraction is not the lever it is often assumed to be.

4.9 Failure Modes

Table 6 shows five representative outputs, one per behaviour discussed above; Appendix C asks where the surrounding text comes from. Content the model has seen in training survives fluently; content it has not seen is either dropped or replaced with a plausible substitute, and the resulting sentence gives no indication that anything is missing.

Two of these behaviours are worth separating from ordinary translation error. The third example is penalised by BLEU despite being an accurate paraphrase, which shows the metric is not uniformly too generous here. The fourth reverses the polarity of the source sentence while remaining fluent and confident, with nothing in the output to signal the reversal to a reader who cannot check the source. Across the test set, 9.9% of positive references are rendered negative, a rate high enough to matter for any deployment where a user cannot verify against the original signing.

5 Conclusion

Cross-lingual transfer improves low-resource sign language translation, and we report the best pose-only result we are aware of on Indian Sign Language from a 15% sample of iSign, though well short of video-based systems on the same corpus. Two findings follow from taking this result apart. First, the benefit comes from a transferred encoder-decoder relationship rather than from either component alone, a pattern robust across every comparison we ran. Second, and independent of any source-language comparison, the system's recall tracks training frequency almost exactly and does not improve on rare or unseen content under any intervention we tested. The source-language advantage also splits unevenly across the body, holding on the hands but not the face.

Limitations

Our conditions run at up to three seeds, which is enough to show when an arm's range is disjoint from another's but not enough to estimate a sampling distribution. One checkpoint per source language remains a single sample of that language, so the 22% encoder advantage is a property of these two checkpoints before it is a property of ASL and CSL.

The ASL and CSL source checkpoints differ in domain as well as sign language. The OpenASL arm weakens a purely domain-based reading, since two ASL corpora of different domains behave alike, but both are still more hand-forward than scripted conversation, so we cannot fully separate the two explanations and present the articulator pattern as suggestive throughout.

Our splits contain residual source-video leakage: 22 videos cross the train–test boundary, contributing 0.90% of test clips. We measure its effect rather than assume it away, and removing the affected clips raises BLEU-4 slightly on every seed, so it is not inflating our results. Signer identity is not distributed with iSign, so signer-level independence cannot be verified directly and source-video separation is the strongest proxy available to us.

Our absolute operating point is low, 3.61 BLEU-4. The decompositions we report are therefore established in a regime where the model translates poorly, and encoder–decoder coupling, frequency-tracking recall, and body-stream dominance may behave differently in a system with more headroom. The vocabulary finding is the one we would most expect to generalise, since it concerns what is absent from the training distribution rather than what the model does with what is present, but we have not tested it elsewhere.

Our non-manual marking experiment is a negative result that admits two readings, as Section 4.7 notes: the substituted eyebrow landmarks reuse weights fitted to jaw-contour statistics, so a structural mismatch and a capacity limit predict the same null.

We study one architecture, one target language, and one transfer direction. The parameter asymmetry between visual and language components is characteristic of gloss-free pose-based sign language translation generally, but the specific face-landmark selection is one design choice among several. The non-manual marking finding in particular should be checked against other systems before being treated as a general property of the architecture class.

Acknowledgments

Generative AI tools were used only for language editing and readability improvements. All research, experimental design, implementation, and results are the authors' own work. We thank the anonymous reviewers, whose questions about reproducibility, split integrity, single-run design, and the interpretation of the vocabulary effect prompted the split audit in Appendix B, the multi-seed tables throughout Section 4, and the language-prior and predictability controls in Section 4.8. Three claims in the submitted version did not survive those checks and have been withdrawn.

References

Keren Artiaga, Sabyasachi Kamila, Haithem Afli, Conor Lynch, and Mohammed Hasanuzzaman. 2025. Rethinking sign language translation: The impact of signer dependence on model evaluation. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 18379–18391.

Keren Artiaga, Yang Li, Ercan Engin Kuruoglu, and Wai Kin Chan. 2024. Cross-sign language transfer learning using domain adaptation with multi-scale temporal alignment. *Multimedia Tools and Applications*, 83(12):37025–37051.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10023–10033.

Anal Roy Chowdhury and Debarshi Kumar Sanyal. 2025. Enhancing Indian Sign Language translation via motion-aware modeling. In *Proceedings of the Workshop on Sign Language Processing (WSLP)*, pages 39–46.

Pedro Dal Bianco, Oscar Stanchi, Facundo Quiroga, and Franco Ronchetti. 2025. Multilingual sign language translation with unified datasets and pose-based transformers. In *Proceedings of the Workshop on Sign Language Processing (WSLP)*, pages 31–36.

Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro-i Nieto. 2021. How2Sign: A large-scale multimodal dataset for continuous American Sign Language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2735–2744.

Shester Gueuwou, Xiaodan Du, Greg Shakhnarovich, Karen Livescu, and Alexander H Liu. 2025. SHuBERT: Self-supervised sign language representation learning via multi-stream cluster prediction. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 28792–28810.

Youngjoon Jang, Haran Raajesh, Liliane Momeni, Gül Varol, and Andrew Zisserman. 2025. Lost in translation, found in context: Sign language translation with contextual cues. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Sheng Jin, Lumin Xu, Jin Xu, Can Wang, Wentao Liu, Chen Qian, Wanli Ouyang, and Ping Luo. 2020. Whole-body human pose estimation in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 196–214.

Abhinav Joshi, Susmit Agrawal, and Ashutosh Modi. 2023. ISLTranslate: Dataset for translating Indian Sign Language. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10466–10475.

Abhinav Joshi, Romit Mohanty, Mounika Kanakanti, Andesha Mangla, Sudeep Choudhary, Monali Barbate, and Ashutosh Modi. 2024. iSign: A benchmark for Indian Sign Language processing. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10827–10844.

Abhinav Joshi, Vaibhav Sharma, Sanjeet Singh, and Ashutosh Modi. 2025. POSESTITCH-SLT: Linguistically inspired pose-stitching for end-to-end sign language translation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 13845–13864.

Zecheng Li, Wengang Zhou, Weichao Zhao, Kepeng Wu, Hezhen Hu, and Houqiang Li. 2025. Uni-Sign: Toward unified sign language understanding at scale. In *International Conference on Learning Representations*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. 2019. MediaPipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191.

Christos Sagonas, Epameinondas Antonakos, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 2016. 300 Faces in-the-wild challenge: database and results. *Image and Vision Computing*, 47:3–18.

Bowen Shi, Diane Brentari, Gregory Shakhnarovich, and Karen Livescu. 2022. Open-domain sign language translation learned from online video. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6365–6379.

Ryan Wong, Necati Cihan Camgoz, and Richard Bowden. 2024. Sign2GPT: Leveraging large language models for gloss-free sign language translation. In *International Conference on Learning Representations*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498.

Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Ulrike Zeshan. 2003. Indo-Pakistani Sign Language grammar: A typological outline. *Sign Language Studies*, 3(2):157–212.

Ulrike Zeshan. 2004. Interrogative constructions in signed languages: Crosslinguistic perspectives. *Language*, 80(1):7–39.

Biao Zhang, Garrett Tanzer, and Orhan Firat. 2024. Scaling sign language translation. In *Advances in Neural Information Processing Systems*, volume 37.

Benjia Zhou, Zhigang Chen, Albert Clapés, Jun Wan, Yanyan Liang, Sergio Escalera, Zhen Lei, and Du Zhang. 2023. Gloss-free sign language translation: Improving from visual-language pretraining. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20871–20881.

Hao Zhou, Wengang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. 2021. Improving sign language translation with monolingual data by sign back-translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1316–1325.

A Training and Checkpoint Details

Source checkpoints. We pretrain nothing. All three sources are the public pose-only translation releases accompanying Li et al. (2025): `how2sign_pose_only_slt.pth`, `openasl_pose_only_slt.pth`, and `csl_daily_pose_only_slt.pth`. They share a stage-1 pose pretraining on CSL-News and differ in the stage-2 translation corpus, so a comparison between two of them isolates stage 2. Each stores 627 tensors: 343 belong to the sign encoder (6.9M parameters) and 284 to the mT5-base decoder, which holds the remainder. Under 1% of the parameters touch the sign input.

Fine-tuning. Every condition trains for 20 epochs on the ISL training split with AdamW (Loshchilov and Hutter, 2019), $\beta = (0.9, 0.98)$, weight decay 0.01 applied to non-bias non-norm parameters only, one-cycle schedule with peak learning rate 3×10^{-4} and 200 warmup steps, label smoothing 0.2, at most 256 frames per clip, and bf16 autocast. Effective batch is 8 throughout; the micro-batch is 2 with gradient accumulation 4 on the 16 GB card and 4 with accumulation 2 on the 24 GB card, which changes sequence padding and is why arms are compared only within a configuration. We take the final epoch: no condition uses development-set checkpoint selection, so none benefits from a model-selection step the others did not receive. Seeds are 13, 21, and 37.

What each condition loads. Every condition instantiates the same architecture and differs only in which tensors are initialised from a source checkpoint and which are trainable (Table 9).

Condition	Loaded	Trained
Encoder + decoder	all 627	all
Encoder only (frozen)	343 encoder	decoder + projection
Decoder only	284 mT5	all
Hand / body / face	that stream's 85 (170 for both hands)	all
Neither (scratch)	none	all
CSL enc + English dec	343 from CSL, 284 from How2Sign	all

Table 9: Parameter initialisation per condition. Tensors not loaded keep the architecture's own pretrained initialisation, which for the decoder is mT5-base (Xue et al., 2021). "Frozen" means the four ST-GCN streams receive no gradient; the projection into the decoder stays trainable, since a frozen projection would confound representation quality with an unadapted interface.

Per-condition chrF. Encoder + decoder 21.80; encoder only 15.02; decoder only 15.50; hand streams only 15.14; body only 15.21; face only 15.39; scratch 15.29. These are means over the same runs as Table 1.

Evaluation. Beam search, 4 beams, 100 new tokens maximum, matching the reference implementation. Scores are computed with `sacrebleu` (Post, 2018) on the full test split. Decoding under bf16 autocast is not bit-deterministic, so a fixed checkpoint varies by up to ± 0.03 BLEU-4 between passes; tables built from one decode pass carry their own control.

Larger-data condition. The data-scale comparison in Section 4.8 uses a second corpus draw of 28,512 training clips and a subsample of it at 19,099, both evaluated on the same 3,534-clip test split with the same split assignment, so only training size varies. The subsample is drawn by source video rather than by clip, so it does not split a video across the two arms. This is a different draw from the 19,093-clip split used everywhere else in the paper, and the two numbers should not be confused.

Configuration across Table 1. The encoder+decoder, frozen-encoder and scratch rows were trained at micro-batch 2 with accumulation 4; the decoder-only and single-stream rows at micro-batch 4 with accumulation 2. Effective batch is 8 in both. Where we can measure the difference, on the ASL arm of Table 2, it is worth $+0.14$ BLEU-4 — in the direction that would *raise* the single-stream rows, and an order of magnitude below the gap those rows are read against.

Provenance. Every run above is recorded with a manifest giving its source checkpoint, loaded tensor count, dataset, seed, and full argument list; these are reproduced from the manifests rather than from the launch scripts. Two exceptions: the manifests for one scratch seed and one frozen-encoder seed were lost with the machine they ran on, and their configuration is taken from the launcher that produced them, whose defaults match the verified manifests of their sibling runs. Both contribute only a second or third seed to a row, never a reported comparison on their own.

B Split Construction and Leakage Audit

iSign clip identifiers follow three conventions, and a rule that handles only the first two silently fragments videos of the third kind into several:

Shape	Video id ends at	Clips
`<id>--<n>`	final `--`	21,243
`<hex>-<n>`	final `-`	2,181
`<id>_<suffix>`	final `_`	497

The third shape carries hyphens inside the identifier itself, for example `Miy-F4TYnso_d`. Our original rule mis-parsed these, which is how 22 source videos came to cross the train–test boundary. Under the corrected rule the three splits hold 3,198, 411, and 413 source videos. Residual leakage is 22 videos and 22 test clips (0.90%); for the data-scale pair it is 39 of 3,534 (1.10%) and 48 of 3,534 (1.36%).

Removing the affected clips changes BLEU-4 by +0.02, +0.03, and +0.02 on seeds 13, 21, and 37 respectively. Scored alone, the leaked clips reach 1.23, 0.69, and 0.73 BLEU-4, well below the full-test-set scores of 3.58, 3.77, and 3.47. The leakage does not inflate any reported number.

C Provenance of Model Output

If the output is not retrieved from training, where does it come from? Comparing output n-grams against the training corpus, and controlling for the fact that real references share that corpus and overlap it by default, hypothesis trigrams appear in training 46.5% of the time against 20.8% for genuine references; at four-grams, 20.2% against 5.6%. The model composes from training material at a higher rate than a real translation does, yet 80% of its four-grams appear nowhere in training: recombination under weak conditioning, not verbatim retrieval.

D Test-Set Anchors

System	BLEU-4	chrF
Constant (commonest train sentence)	0.00	0.21
Random training sentence	0.06	12.63
Length-matched random sentence	0.15	15.14
Scratch (trained, no transfer)	0.46	15.29
ASL transfer (ours)	3.61	21.80

ASL transfer is a three-seed mean and scratch a two-seed mean, matching Table 1; both are rounded to two decimals here.

E Full Articulator Masking Lattice

All sixteen subsets, seed-13 checkpoint, one decode pass. Date recall is over the 111 test clips whose reference contains a date.

Streams retained	BLEU-4	chrF	Date
body + left + right + face	3.61	21.91	68.5%
body + right + face	2.90	20.56	62.2%
body + left + right	2.08	19.45	60.4%
body + right	1.26	17.42	55.0%
body + left + face	1.09	15.61	0.0%
left + right + face	0.47	15.60	27.9%
body + face	0.24	13.54	0.0%
body + left	0.22	13.83	0.0%
right + face	0.21	14.99	15.3%
left + right	0.12	14.25	0.9%
right	0.10	13.97	0.0%
body	0.05	13.42	0.0%
left + face	0.05	14.50	0.0%
left	0.03	14.05	0.0%
face	0.03	13.73	0.0%
none (blind)	0.03	13.96	0.0%

At sizes two and three, every subset containing body outperforms every same-sized subset without it. At size one the ordering reverses: body alone (0.05) is indistinguishable from blind decoding while the right hand alone reaches 0.10, which is the not-sufficient half of the claim in Section 4.6. Note also that the four subsets recovering more than half of dates all contain body together with the right hand.

WSLP-Task2 at WSLP@EMNLP 2026: Gated Vision–Language Adjudication and Multi-Expert Fusion for Isolated Sign Language Recognition

Bolisetty Samanvitha and Ojas Srivastava
Pruthwik Mishra and Praveen Kumar Chandaliya
Sardar Vallabhbhai National Institute of Technology (SVNIT), Surat, India
{u24ai024, u24ai028, pruthwikmishra, pkc}@aid.svnit.ac.in

Abstract

We present WSLP-Task2 for the WSLP@EMNLP 2026 isolated sign language recognition shared task. Given a short Indian Sign Language video, the system must return a closed-vocabulary ranking of ten glosses. With 4,398 training clips spanning 4,361 glosses (4,336 singleton classes), softmax classification is ill-posed; we therefore cast experts as metric learners with gallery retrieval, or as a gated vision–language decoder. Our approach combines a fine-tuned I3D–DINO RGB ensemble (64.45% Top-1), a Qwen2.5-VL-3B LoRA adjudicator restricted to disagreement or low-margin clips, and auxiliary DINOv2–ArcFace, ViTPose–BiLSTM, Pose-TCN, and MambaVision–CosFace experts. Late fusion updates the top-1 prediction only when independent policies agree, reciprocal rank fusion selects a candidate already supported by multiple variants, or a filtered VLM vote lies inside the RGB shortlist. On the blind Codabench test set (n=526), ensemble reciprocal rank fusion attains 66.16% Top-1 (72.43% Top-5, 73.19% Top-10). The improvement over the disagreement-gated RGB ensemble (65.97%) is only 0.19 percentage points, corresponding to one additional correct prediction on the 526-clip test set. We therefore interpret the main finding as evidence that selective and constrained expert intervention can preserve the strong RGB baseline while avoiding the degradation caused by unrestricted fusion.

1 Introduction

Isolated sign language recognition (ISLR) maps a short video of a single sign to an English gloss. WSLP@EMNLP 2026 Task 2 evaluates Top-1 / Top-5 / Top-10 accuracy on 526 blind test clips under a closed training vocabulary (Exploration Lab, IIT Kanpur, 2026). Two dataset properties dominate system design. First, the training split is nearly one-shot: 4,398 clips over 4,361 glosses, so conventional softmax classifiers overfit quickly. Second, RGB convolutional ensembles already reach the mid-60s: a frozen I3D–DINO–EfficientNet–SegFormer ensemble scored 62.36%, and fine-tuning I3D with ArcFace raised the same four-way ensemble to 64.45% (Carreira and Zisserman, 2017; Deng et al., 2019).

The remaining Top-1 errors concentrate where RGB experts disagree. On a disagreement-gated RGB ensemble (65.97% Codabench), about 260 clips have fine-tuned I3D and EfficientNet predictions that conflict; accuracy on this subset is only 55–58%. We therefore treat vision–language and metric-learning models as *auxiliary adjudicators*, not as replacements for the RGB ensemble: Qwen2.5-VL-3B LoRA (Bai et al., 2025; Hu et al., 2022) is invoked only under disagreement or low confidence margin; DINOv2–ArcFace, ViTPose–BiLSTM, and related experts may revise the top-1 gloss only when their candidate already appears in the RGB shortlist.

Our contributions are: (1) a selective expert-fusion framework in which a strong RGB ensemble serves as the default predictor and auxiliary experts are allowed to intervene only under disagreement or low-confidence conditions; (2) a systematic evaluation of constrained late-fusion strategies combining appearance, pose, and vision–language evidence; and (3) an empirical analysis showing that selective, high-precision expert intervention is more reliable than unrestricted fusion in the near-one-shot setting.

2 Related Work

Word-level sign recognition commonly uses I3D on RGB (Carreira and Zisserman, 2017; Li et al., 2020) and self-supervised appearance features such as DINOv2 (Oquab et al., 2024). Pose streams (MediaPipe Holistic / ViTPose with temporal heads) complement RGB (Xu et al., 2022; WSLP Or-

Proceedings of the Second Workshop on Sign Language Processing, pages 237–242
October 29, 2026 ©2026 Association for Computational Linguistics

ganizers, 2025). ArcFace and CosFace provide additive-margin metric learning for closed-set retrieval (Deng et al., 2019; Wang et al., 2018). We adapt a vision–language model with LoRA as a gated adjudicator rather than an open-vocabulary captioner, and combine ranked lists with reciprocal rank fusion (Cormack et al., 2009).

3 Task and Data

Task. For each test clip, systems submit a closed-vocabulary ranking of ten glosses. Codabench reports Top-1/5/10 accuracy on n=526 clips.

Data. We use the official WSLP word-recognition videos and splits. CISLR clips that overlap the WSLP vocabulary are relabelled to form an 8,154-clip augmented training set for Qwen, DINOv2, and ViTPose. Offline leave-one-out nearest-neighbor probes on multi-clip classes are used only for development diagnostics and are never reported as leaderboard scores.

4 Method

Figure 1 summarizes the pipeline. All late-fusion variants start from a disagreement-gated RGB ensemble and revise the top-1 prediction only under the gates below.

Design principle. The framework follows an asymmetric intervention principle: the primary RGB ensemble is trusted by default, while auxiliary experts are treated as sources of corrective evidence rather than independent classifiers. An auxiliary prediction is therefore permitted to replace the primary top-1 prediction only when the primary models indicate uncertainty and the candidate satisfies additional shortlist or agreement constraints. This design reflects the strong performance of the RGB ensemble and the substantially weaker standalone performance of several auxiliary experts.

4.1 Training objectives

Let $z = f_\theta(x) \in \mathbb{R}^d$ be an ℓ_2-normalized embedding of clip x. Gallery retrieval ranks gloss g by $\mathrm{sim}(z, e_g) = z^\top e_g$.

ArcFace (fine-tuned I3D / DINOv2). With class weights $W \in \mathbb{R}^{|G|\times d}$, scale s, margin m:

$$\theta_j = \arccos\big(\mathrm{clip}(z^\top W_j, -1+\epsilon, 1-\epsilon)\big), \quad (1)$$

$$\mathcal{L}_{\text{ArcFace}} = -\log \frac{\exp(s\cos(\theta_y + m))}{\sum_j \exp\big(s\cos(\theta_j + m\cdot \mathbb{1}[j=y])\big)}. \quad (2)$$

FT-I3D heads use s=30, m=0.30, embedding dimension 256. DINOv2 ArcFace uses s=20, m=0.20, emb 256.

CosFace (MambaVision).

$$\mathcal{L}_{\text{CosFace}} = -\log \frac{\exp\big(s(\cos\theta_y - m)\big)}{\sum_j \exp(s\cos\theta_j)}. \quad (3)$$

Qwen gloss LM. LoRA-adapted Qwen2.5-VL minimizes causal cross-entropy on gold gloss tokens:

$$\mathcal{L}_{\text{VLM}} = -\sum_{t=1}^{T} \log p_\phi(w_t \mid w_{<t}, x). \quad (4)$$

Confidence gate. With expert scores $s_1 \geq s_2$, define $m = s_1 - s_2$. Qwen runs iff

$$\begin{aligned}\mathrm{gate}(x) = &\mathbb{1}[\hat{y}_{\text{I3D}} \neq \hat{y}_{\text{DINO}}] \\ &\vee \min(m_{\text{I3D}}, m_{\text{DINO}}) < \tau,\end{aligned} \quad (5)$$

with τ=0.05; otherwise the stronger RGB ranking is retained (Figure 1).

Confidence-aware reciprocal rank fusion. With k_0=60,

$$\mathrm{RRF}_w(w) = \sum_k \frac{w_k}{k_0 + \mathrm{rank}_k(w)}. \quad (6)$$

Easy clips (I3D=DINO and both margins >0.05) use weights $(0.55, 0.45, 0)$ for (I3D, DINO, Qwen), or a 0.1 Qwen drip if a Qwen row exists. Hard gated clips use $(0.45, 0.20, 0.35)$ or $(0.20, 0.45, 0.35)$, favoring the stronger RGB expert plus Qwen.

Weighted Borda.

$$B(g) = \sum_k w_k \cdot \max(0,\ 10 - \mathrm{rank}_k(g)). \quad (7)$$

4.2 RGB ensemble

A frozen four-way reciprocal rank fusion (RRF) over I3D, DINOv2, EfficientNet-B4, and SegFormer reaches 62.36% Top-1. Fine-tuning I3D and retraining ArcFace heads raises the same ensemble to 64.45% (+2.09pp). Locked stream weights (k=60) are I3D:DINOv2:EfficientNet:SegFormer = 1.30:0.40:1.20:0.35. Within the hybrid

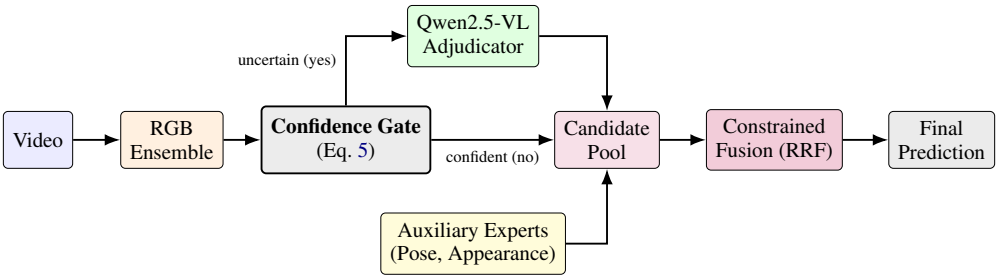


Figure 1: WSLP-Task2 pipeline. The primary RGB ensemble processes every video. The confidence gate invokes the Qwen2.5-VL adjudicator only for uncertain predictions. Auxiliary experts contribute candidates to the constrained late fusion, which restricts updates to preserve reliable RGB predictions.

I3D–DINOv2 expert, gallery and class logits are mixed at 0.7:0.3 and then combined with equal z-normalized weights. Single-stream RGB baselines score only about 52%. Disagreement-gated refinement of I3D–EfficientNet conflicts, restricted to Top-5 candidates, yields a 65.97% RGB ensemble that anchors all subsequent fusion variants.

4.3 Qwen2.5-VL LoRA adjudicator

Adapters. We fine-tune Qwen2.5-VL-3B-Instruct with LoRA. The prompt-augmented adapter uses rank r=64, α=128, dropout 0.05 on attention and MLP projections; 16 frames; 8,000 steps; learning rate 8×10^{-5}; and six prompt paraphrases. A second adapter freezes the bottom eight LLM layers and the vision tower (r=32, α=64, 6,000 steps, learning rate 10^{-4}) on the same augmented CISLR+WSLP data. Additional adapters with varied freezing and capacity supply gated predictions to agreement-based and Borda fusion only.

Prompts and decoding. At inference the model is instructed to output a single English gloss for an Indian Sign Language clip. Training samples uniformly from six paraphrases with fixed gold targets. A pairwise variant asks which of two RGB Top-5 glosses better matches the video (answer A or B). Generated text is normalized and mapped onto the closed WSLP vocabulary by exact, prefix, and substring matching; unmapped strings are discarded.

Disagreement diagnostic. On the 260 clips where I3D and DINOv2 disagree, ungated Qwen predictions differ from the RGB top-1 on 255 clips, enter the RGB Top-10 on only 8, and agree with I3D/EfficientNet on only 2. Ungated substitution is therefore harmful; agreement-based fusion admits at most five Qwen votes that already lie in the RGB Top-6 or side with I3D/EfficientNet.

4.4 Auxiliary metric and pose experts

DINOv2–ArcFace. We apply LoRA (r=16, α=32) to DINOv2-base query/value projections with an ArcFace head (s=20, m=0.2, embedding size 256), using six frames per clip for 30 epochs on 8,154 WSLP+CISLR clips. Leave-one-out retrieval reaches 0.788 offline, but ungated Codabench Top-1 is only ≈0.30; gated fusion reaches 0.6426. Agreement with the 65.97% RGB ensemble is 30.4% at top-1 and 51% within Top-5, so DINOv2 is used only as a constrained candidate.

ViTPose–BiLSTM. Pooled ViTPose-plus-large embeddings feed a BiLSTM with ArcFace under temporal augmentation. Early pose-only systems collapsed to the majority gloss “Pencil”; after ArcFace training, leave-one-out accuracy is 0.393 with 463 distinct top-1 predictions. A gated ViTPose variant matches the 65.97% RGB ensemble with five updates. Floor agreement is only ≈5.7%, so constrained fusion ignores Pencil votes and Borda down-weights pose-embedding streams to 0.4/0.3.

Pose-TCN. MediaPipe Holistic features (225-D) with a temporal convolutional head enter Borda at weight 0.9. As a single model, Pose-TCN scores 0.1103 Top-1.

MambaVision–CosFace. Partial fine-tuning with CosFace yields offline probes near 0.42, but raw Codabench Top-1 is 0.1179; fused Mamba variants reach 0.616–0.627. Multi-feature embedding fusion including MambaVision scores 0.6407.

Negative results. Gemma-4 E2B LoRA collapses to a few majority glosses and is excluded

from fusion. Frozen-backbone DINOv2 LoRA probes remain at 2–3% and are discarded.

4.5 Late fusion strategies

Our fusion pipeline utilizes 15 constrained variants combined via RRF. The expert weights, disagreement thresholds (τ), shortlist constraints, and update caps (d) were selected using only development-time leave-one-out nearest-neighbor diagnostics available before blind-test evaluation. The blind test set was used only for final evaluation. These settings are therefore task-specific and should not be interpreted as dataset-independent hyperparameters. Our goal is to evaluate the selective-intervention principle under the competition's near-one-shot setting rather than claim direct transfer of the complete fusion configuration to other datasets.

Unless noted, variants start from the disagreement-gated RGB ensemble (0.6597). Let d denote the fraction of clips whose top-1 prediction changes. We group the strategies into three core mechanisms:

Constrained Multi-Expert & Stream-Level RRF. We enforce that candidate updates only occur when primary vision models (I3D and EfficientNet) disagree. Candidates must already reside within the RGB Top-5 or Top-7 to be considered valid, preventing weaker experts from injecting out-of-context glosses. Updates are strictly capped (e.g., $d \leq 0.15$ to 0.22).

Weighted Borda & Agreement Fusion. These variants assign offline-derived weights to each stream (e.g., RGB=1.2, Pose=0.9). We replace the RGB top-1 prediction only when a strong consensus emerges: either $\geq$3 auxiliary experts agree, or the Borda score exceeds the RGB score by at least 15% within the Top-8 shortlist.

Ensemble RRF (final system; d=0.0361). RRF over fifteen constrained fusion variants, with higher weight on the disagreement-gated RGB ensemble (w=1.5) and moderate weight on ViTPose- and Qwen-supported variants (w=0.8). Adopt a fused top-1 only if at least two variants already support it; cap $d \leq 0.20$. This is our reported system: 0.6616 Top-1, 0.7243 Top-5, 0.7319 Top-10.

Unsuccessful combinations. Constrained multi-expert fusion with gated DINOv2 scores 0.6350. Joint DINOv2, freeze-8 Qwen, and ViTPose scores 0.6407. Both show that strong offline experts can degrade Top-1 without multi-variant consensus.

5 Experimental Setup

Experiments run on the SVNIT Aurora GPU cluster with offline Hugging Face caches on compute nodes. Table 1 summarizes model objectives and key hyperparameters.

Model	Objective	Key settings
Fine-tuned I3D	ArcFace (s=30, m=0.30)	Four-way ens. 64.45%
Qwen2.5-VL (prompt-aug)	Causal CE	16 frames; LoRA r=64
Qwen2.5-VL (freeze-8)	Causal CE	LoRA r=32; freeze 8
DINOv2–ArcFace	ArcFace (s=20, m=0.2)	LoRA r=16; LOO 0.788
ViTPose–BiLSTM	ArcFace	LOO 0.393; gated 0.6597
MambaVision	CosFace	Single-model 0.1179

Table 1: Optimization settings for the reported models.

6 Results

Table 2 reports Codabench Top-1 accuracy. Our public entry ranked fifth at 66.16 / 72.43 / 73.19 (Top-1/5/10). Leading systems reach 72.05% Top-1 while Top-5/10 are already close to ours, suggesting that shortlist coverage is strong and top-1 selection remains the bottleneck.

7 Analysis

Statistical Significance and the Fusion Gap. The final 15-variant Ensemble RRF achieves 0.6616, improving over the disagreement-gated RGB ensemble's 0.6597 by exactly one additional correct prediction out of 526 test clips. A paired McNemar test does not indicate a statistically significant difference ($p > 0.05$). Consequently, we do not interpret the 0.19-point increase as evidence of a substantial accuracy improvement. Instead, our core finding is that disagreement-gated constraints successfully prevent the performance degradation that occurs with unrestricted fusion in this low-resource setting. Without these constraints, incorporating weaker experts substantially harms Top-1 accuracy.

Expert Ablation. To understand which auxiliary experts contribute to the final 0.6616 Ensemble RRF, we perform a leave-one-out ablation on the fusion pool. Removing the Qwen adjudicator variants reduces performance to 0.6559, while removing

#	System	Top-1
1	Ensemble RRF (final system)	0.6616
2	Disagreement-gated RGB ensemble	0.6597
3	Constrained multi-expert fusion	0.6559
4	Confidence-weighted stream RRF	0.6540
5	Low-edit refinement	0.6521
6	Consensus collision resolution	0.6502
7	Fine-tuned I3D four-way ensemble	0.6445
8	Equal-weight Top-5 consensus	0.6445
9	Gated DINOv2–ArcFace	0.6426
10	Multi-expert DINOv2–Qwen–ViTPose	0.6407
11	Constrained fusion + DINOv2	0.6350
12	High-edit fusion	0.6274
13	Pairwise VLM ranking	0.6103
14	Soft Borda consensus	0.5951
15	Single-model DINOv2 / Mamba / Pose-TCN	0.2985 / 0.1179 / 0.1103

Table 2: Codabench Top-1 on the blind test set (n=526).

Setting	Rule	Outcome
Confidence gate	Eq. 5; ∼260 clips	Agreement fusion
Ungated VLM substitution	Replace RGB top-1	255/260 outside Top-10
Gated VLM votes	Top-6 or I3D/Eff agree; ≤5	5 accepted updates
Pairwise VLM ranking	RGB Top-5; wins ≥4	Below 0.6616
Adapter variants	Gated predictions only	Agreement / Borda

Table 3: Ablations of the Qwen2.5-VL adjudicator.

Expert	w_k	Type
FT-I3D / RGB ens.	1.2	RGB
EffNet / I3D–DINO / Qwen	1.0	RGB/VLM
Pose-TCN	0.9	Pose
DINOv2 / SegFormer	0.8	App.
Light Qwen adapter	0.7	VLM
Pose keypoints	0.5	Pose
ViTPose LSTM/TF	0.4/0.3	Pose

Table 4: Borda weights (Eq. 7).

ViTPose and DINOv2 appearance variants gives 0.6578 and 0.6597, respectively. These results suggest that the final fusion draws complementary evidence from multiple expert families, although the absolute differences are small and should not be interpreted as statistically reliable gains for any individual expert family.

Errors concentrate on disagreement clips. The disagreement-gated RGB ensemble is already correct on most easy clips. Further gains require changing incorrect conflict cases with high per-update precision. Ensemble RRF revises only 19/526 predictions (d=0.0361) yet yields the best score; stream-level RRF revises 115 (d=0.2186) and is less reliable.

Vision–language models require gating. On the 260-clip disagreement set, ungated Qwen almost always proposes a gloss outside the RGB shortlist. Pairwise ranking inside the RGB Top-5 is safer than free generation, but combinations that promote on Qwen∩DINOv2 agreement still fall below 0.6616.

Offline probes overestimate leaderboard gains. DINOv2 leave-one-out 0.788 collapses to ∼0.30 alone and ∼0.64 when gated. ViTPose leave-one-out 0.393 coexists with a 0.6597 Codabench score only because ViTPose is embedded in the RGB ensemble, not because the probe predicts leaderboard accuracy. MambaVision and Pose-TCN show the same pattern (0.12 / 0.11 alone).

Update rate versus accuracy. Preserving the RGB ensemble without further updates retains 0.6616 when used inside ensemble RRF. Equal-weight Top-5 consensus (16 updates) scores 0.6445; soft Borda (71) scores 0.5951; collision resolution scores 0.6502; high-edit fusion (28) scores 0.6274; pairwise VLM ranking scores 0.6103. Constrained fusion with DINOv2 and joint DINOv2–Qwen–ViTPose score 0.6350 and 0.6407—both below the final system. Equal-weight voting fails because the RGB ensemble already matches the final top-1 on 84% of clips, whereas DINOv2 matches on only 37%. Confidence-weighted stream RRF (0.6540) improves on equal consensus but still trails the final system.

Mode collapse. Ungated ViTPose systems predicted "Pencil" for nearly all 526 clips. Gemma collapsed to a few majority glosses. Auxiliary pose and VLM experts are therefore useful only under constraints, never as standalone systems.

Top-5 versus Top-1 and Error Analysis. With Top-5 already near the public leader, roughly thirty clips contain the correct gloss in the Top-5 but not at rank one. Error analysis on these 30 misranked clips reveals three primary failure modes: (1) fine-grained handshape confusion on visually similar signs (e.g., distinguishing *Apple* vs *Onion* where motion is identical but finger closure differs); (2) severe motion blur during the critical apex frame, causing the RGB models to rely on context; and (3) out-of-distribution signers not well-represented in the near one-shot training set. Future improvements will require dedicated high-resolution hand-crop models and better temporal alignment, rather than simply re-ranking existing holistic features.

8 Conclusion

We presented WSLP-Task2, a selective multi-expert framework for near-one-shot isolated Indian Sign Language recognition. The framework uses an I3D–DINOv2 RGB ensemble as the primary predictor and allows vision–language, appearance, and pose experts to intervene only when the primary prediction is uncertain and additional agreement or shortlist constraints are satisfied.

The final Ensemble RRF achieves 66.16% Top-1, 72.43% Top-5, and 73.19% Top-10 accuracy on the blind test set. Although this improves over the 65.97% disagreement-gated RGB ensemble by only 0.19 percentage points and is not statistically significant, the experiments consistently show that unrestricted expert fusion can substantially degrade performance. The main empirical finding is therefore that selective, high-precision expert intervention is more reliable than allowing heterogeneous experts to modify predictions freely.

Error analysis indicates that further gains are likely to require stronger fine-grained hand representations, improved temporal alignment, and better handling of signer variation. More broadly, our results suggest that in near-one-shot recognition, effective expert coordination may be more important than simply increasing the number of models.

References

Shuai Bai, Keqin Chen, Xuejing Liu, and 1 others. 2025. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*.

Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*.

Gordon V. Cormack, Charles L. A. Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *SIGIR*.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. 2019. ArcFace: Additive angular margin loss for deep face recognition. In *CVPR*.

Exploration Lab, IIT Kanpur. 2026. WSLP@EMNLP 2026 shared task: Isolated sign language recognition.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *ICLR*.

Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. 2020. WLASL: A large-scale dataset for word-level american sign language. In *ECCV*.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, and 1 others. 2024. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*.

Hao Wang, Yitong Wang, Zheng Zhou, Shiguang Ji, Dihong Gong, Jingdong Zhou, Zhifeng Li, and Wei Liu. 2018. CosFace: Large margin cosine loss for deep face recognition. In *CVPR*.

WSLP Organizers. 2025. Pose-TCN baselines for word-level sign language recognition (WSLP 2025). Workshop baseline; MediaPipe Holistic 225-D + TCN.

Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. 2022. ViTPose: Simple vision transformer baselines for human pose estimation. In *NeurIPS*.

Testing Video MLLM Sensitivity to Hand-Localized Visual Detail in Chinese Sign Recognition: A Matched Visual Intervention Study

Bowen Ye[1] **Dengfeng Yao**[1,2,3] **Siyuan Jing**[2]
Peng Jin[2] **Hongzhe Liu**[1] **Cheng Xu**[1]

[1] Beijing Key Laboratory of Information Service Engineering, Beijing Union University, Beijing, China
[3] Lab of Computational Linguistics, School of Humanities, Tsinghua University, Beijing, China
[2] Sichuan Province Key Laboratory of Philosophy and Social Science for Language Intelligence in Special Education, Leshan Normal University, Leshan 614000, China
Corresponding author: Dengfeng Yao (tjtdengfeng@buu.edu.cn)

Abstract

Recent benchmarks show that multimodal large language models (MLLMs) can answer questions about sign-language video, but accuracy alone does not reveal which visual cues support those answers. We test behavioral sensitivity to hand-localized visual detail using 150 front-view isolated signs from NationalCSL-DP. For each item, a hand-target condition (T) blurs tracked hand regions and a same-arm control (C) applies the same Gaussian operator, nominal σ, and translated mask geometry to nearby arm regions; the underlying regional content is not information-matched. The same four-choice question is used across conditions and four video MLLMs. All 1,800 final Primary responses are uniquely scorable, with 1,200 T/C responses entering the primary contrast. Mean accuracy is 38.17% on original videos (O), 33.50% on T, and 36.67% on C. The pre-specified C–T contrast is +3.17 percentage points (95% bootstrap CI $[-0.17, 6.50]$; item-cluster sign-flip $p = 0.08172$). The four model-level point estimates are descriptively positive, but two models are floor-limited. The evidence is therefore directional rather than confirmatory. The contribution is a matched-control diagnostic that complements aggregate benchmark accuracy while making its residual anatomical-content mismatch explicit.

1 Introduction

Signed languages are full natural languages whose lexical and grammatical contrasts are expressed through coordinated manual and non-manual signals. For isolated signs, fine hand appearance—including handshape and local orientation cues—can be central, while a video model can also exploit easier regularities in the signer, body, background, or overall motion. A correct multiple-choice answer therefore does not by itself show that the model was sensitive to the visual detail that linguistically distinguishes the sign.

CNSL-bench evaluates 21 open- and closed-source MLLMs on Chinese National Sign Language and documents large capability gaps across modalities and articulatory categories (Zhao et al., 2026). Recent sign-language MLLM work also spans linguistically informed modeling and small-VLM translation (Inan et al., 2025; Chowdhury and Sanyal, 2026). These results establish what systems can answer, but leave a narrower question open: are video-MLLM decisions selectively sensitive to fine visual detail localized on the hands?

We test that question by perturbing the input rather than changing the task. Each sign is rendered in three conditions: an unmodified standardized video (O), a hand-target condition (T), and a same-arm control (C). T and C use the same Gaussian operator and nominal severity, but at different anatomical locations (Figure 1). If hand-localized detail matters more than a nearby local perturbation, accuracy should be lower under T than C. This is a behavioral diagnostic, not a claim to recover an internal causal representation.

The study makes three contributions. First, it turns isolated-sign recognition into a cue-sensitivity diagnostic rather than another model-ranking exercise. Second, it introduces an anatomically constrained translated control with a pairwise-symmetric no-op rule. Third, it evaluates a frozen four-model panel with pre-specified paired and item-cluster inference. The Primary estimate is +3.17 points, but its interval includes zero and the sign-flip test remains above 0.05; we treat that uncertainty as part of the result rather than as something to optimize away.

Proceedings of the Second Workshop on Sign Language Processing, pages 243–248
October 29, 2026 ©2026 Association for Computational Linguistics

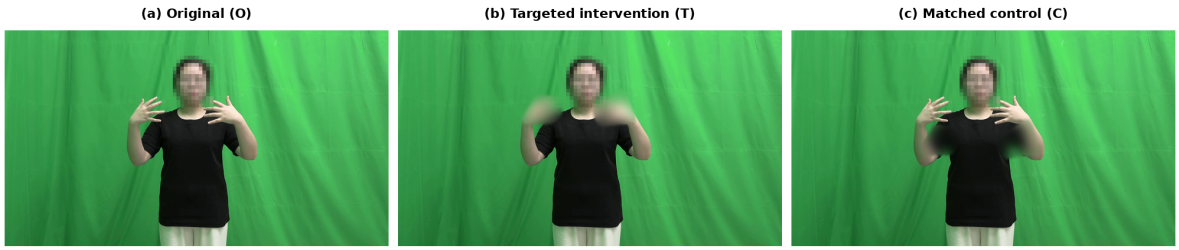


Figure 1: A frame from one of the 150 evaluation items. O is the standardized input; T applies Gaussian blur to tracked hand regions; C applies the same local operator to a translated copy of the target envelope along the same arm. The face is pixelated only in this publication figure; model inputs were unchanged.

2 Related Work

Signed languages should not be reduced to generic gesture streams (Yin et al., 2021). Sign-language MLLM studies expose modeling, data, and evaluation challenges, including signer-dependence effects (Inan et al., 2025; Chowdhury and Sanyal, 2026; Artiaga et al., 2025). Broader video-MLLM diagnostics reveal temporal and motion-specific weaknesses through MVBench, TempCompass, Video-MME, and MotionBench (Li et al., 2024; Liu et al., 2024; Fu et al., 2025; Hong et al., 2025). Together, they motivate diagnostics of what visual-temporal information supports a decision, rather than accuracy alone.

Our closest benchmark context is CNSL-bench (Zhao et al., 2026). We do not extend its leaderboard; we hold the item and question fixed and locally alter a hypothesized evidence source. Related to contrast-set evaluation (Gardner et al., 2020), our change is label-preserving and anatomically paired: the scientific quantity is the *difference between two local degradations*, not the drop caused by blur alone.

3 Matched Visual Intervention

Data, sampling, and task. NationalCSL-DP contains 6,707 Chinese National Sign Language signs and 134,140 dual-view videos from ten signers (Jin et al., 2025). We use Participant 01's front-view stream, making this a controlled within-signer diagnostic rather than a signer-generalization test. The pre-outcome protocol removed explicit variant-suffix entries and three media-only calibration signs, leaving 6,026 eligible entries. A frozen PCG64 draw selected 180 targets; the first 150 were FORMAL and the remaining 30 RESERVE. All source videos passed the media gate, so no reserve item was used. Each four-choice item pairs the target with three random distractors from the same eligible pool under frozen identity and translation-uniqueness constraints; distractors were not semantic nearest neighbors. Candidate sets and order are identical across conditions and models, with correct positions A=38, B=38, C=37, D=37. Dataset-provided lexical labels are used as the answer key and were not independently re-annotated for this study. Exact construction seeds are in Appendix A.

Design factor	Control and residual scope
Local corruption	Same Gaussian operator and nominal σ; regional content is not information-matched.
Mask geometry	Translate the target envelope; anatomical texture and motion may still differ by location.
Question/options	Fix prompt, candidate set, and option order within each item.
Invalid placement	Apply PAIRWISE-SYMMETRIC-NOOP to both T and C for that hand-frame.
Outcome selection	Freeze items/media before scoring; no outcome-based replacement.

Table 1: What the C–T design controls, and what remains residual.

Conditions and identification logic. O is the standardized source video. T applies Gaussian low-pass blur to tracked hand regions. C translates a copy of the same hand-shaped mask along the corresponding wrist–elbow–shoulder path and applies the same operator and nominal σ there (Figure 1). Thus T and C share the video, question, candidate set, mask geometry, operator, and nominal severity; they do *not* guarantee equal perceptual or model-relevant information removal because the underlying regional content differs. A positive $\mathrm{Acc}(C) - \mathrm{Acc}(T)$ therefore indicates greater behavioral sensitivity to the hand-localized placement under this matched-corruption design, not pure removal of linguistic information. O is contextual; O–T is not the primary estimand.

Geometry and degradation. Hand and arm locations are obtained with RTMW-L (Jiang et al., 2024) at 384×288, followed by a five-frame temporal median. The 21 hand keypoints define a convex-hull core; the target envelope dilates it by $0.15\times$ hand scale. Candidate controls translate a copy of this envelope along the same-arm path with $\gamma \in \{1.0, 1.1, \ldots, 3.0\}$, selecting the smallest valid placement with hand-core overlap ≤ 0.05, target/control IoU ≤ 0.10, and area retention ≥ 0.98.

Both T and C use Gaussian blur with $\sigma = 0.16\times$ hand scale and boundary feathering with $\sigma = 0.04\times$ hand scale. If no valid control exists for a hand-frame, that hand-frame is left unchanged in both T and C (PAIRWISE-SYMMETRIC-NOOP). Formal items require at least 98% pairwise-valid hand-frame coverage and no invalid interval longer than 0.15 s. In the final set, 55,172/55,248 hand-frames (99.8624%) are pairwise-valid; 76 use symmetric no-op. Minimum item coverage is 98.5119%, and the longest invalid interval is 0.12 s.

Pre-outcome media safeguards. Severity was selected only from fixed media calibration. At $\sigma = 0.16\times$ hand scale, fine finger separation and local handshape edges were suppressed while coarse hand position and gross trajectory remained readable. A pre-outcome calibration-panel MAE sanity check was approximately 0.1845 for T versus 0.1698 for C; it was not an information-equivalence metric and motivates treating the corruption as nominally, not information-, matched. All 450 formal videos passed full decode/resolution/frame-count/frame-rate checks; O/T/C keep the same native rate (25 or 50 fps) within an item and 1920×1080 resolution. No outcome-based replacement occurred.

4 Experimental Setup and Inference

Prompt and execution. The frozen Chinese prompt presents four candidate glosses and requests one letter (A/B/C/D) with no explanation; exact wording and manifest hash are in Appendix A. Prompt bytes and option order are identical for O/T/C and across providers. The execution manifest is blinded to the correct letter and target gloss; the scoring key is separately sealed. Condition order is balanced across all six O/T/C permutations (25 items each), and each provider follows a pre-generated item order with an item's three conditions adjacent.

The completed Primary panel contains Qwen, GLM, Doubao, and ERNIE; each receives all 150 items in all three conditions. All 1,800 final Primary responses are uniquely scorable (malformed count: 0), with 1,200 T/C responses entering the estimand. Each scheduled call seeks one scientific response; retries are infrastructure-only, never quality resampling. Frozen decoding/request details are in Appendix A. Provider video ingestion is partly unobservable, so absolute accuracies are descriptive rather than a cross-provider leaderboard.

Gemini and MiniCPM-V were predeclared secondary systems but were not closed before the scoring key was opened, so later secondary execution is recorded as a sequence deviation. Gemini remained partial after deblinding (20 valid responses; the rest infrastructure-failed or not run), and MiniCPM-V never reached formal video execution; neither altered Primary requests, panel membership, or statistics.

Inference. The primary estimand is $\Delta = \mathrm{Acc}(C) - \mathrm{Acc}(T)$. Per-model paired comparisons use exact McNemar tests with Holm adjustment across four tests. For each item, d_i is the four-model mean of $\mathrm{correct}(C) - \mathrm{correct}(T)$. The confirmatory panel-level test flips the sign of d_i at the item-cluster level (100,000 Monte Carlo flips; seed 20260819031). Under the null, these item-level signed contrasts are treated as sign-symmetric around zero for this fixed four-model panel; inference therefore targets the sampled items under this panel, not a population of video MLLMs. A 10,000-resample item-cluster percentile bootstrap (seed 20260819032) gives the 95% interval. A pre-specified floor guard flags models with $\mathrm{Acc}(O) \leq 0.30$; flagged models remain equally weighted in the fixed panel.

5 Results

All 150 items enter the panel-level analysis, and all 600 model-by-item T/C pairs are complete. Mean accuracy is 38.17% on O, 33.50% on T, and 36.67% on C. Relative to O, T lowers mean accuracy by 4.67 points and C by 1.50. The Primary C–T estimate is +3.17 points, with a bootstrap 95% interval of $[-0.17, +6.50]$ and sign-flip $p = 0.08172$; the interval includes zero and the confirmatory test does not cross 0.05.

Descriptively, C–T is positive for each model (Table 2). C-correct/T-wrong versus T-correct/C-wrong counts are 16/7 for Qwen, 5/3 for GLM,

Model	O	T	C	C–T
Qwen	46.67	37.33	43.33	+6.00
GLM	28.00	27.33	28.67	+1.33
Doubao	48.00	44.00	47.33	+3.33
ERNIE	30.00	25.33	27.33	+2.00

Table 2: Primary results. O/T/C are accuracies (%); C–T is in percentage points. GLM and ERNIE trigger the pre-specified floor guard.

26/21 for Doubao, and 8/5 for ERNIE; unadjusted exact McNemar p values are 0.09314, 0.72656, 0.56006, and 0.58105, and none survives Holm adjustment. The interval and panel-level sign-flip result are the primary evidence; the four same-sign point estimates are descriptive context only. We therefore treat the result as directional, not confirmed.

6 Discussion

The C–T comparison is stronger than a clean-versus-blurred ablation because it holds the local Gaussian operator, nominal σ, mask construction, question, and options fixed while moving the perturbation anatomically. It does not, however, make the two regions information-equivalent. Hands contain different spatial detail and motion structure from nearby forearm regions, and the pre-outcome media sanity check itself shows a modest T>C pixel-difference asymmetry. A generic sensitivity to the amount of local detail removed can therefore contribute to C–T. The estimand is best read as selective behavioral sensitivity to the hand-localized placement under a nominally matched corruption, not as a pure measure of linguistic evidence use.

The +3.17-point estimate is uncertain: its 95% interval crosses zero and two models are floor-limited near the 25% chance level. The fixed-panel test and interval are more informative than sign agreement across four point estimates. Retaining the floor-limited models is deliberate: the Primary estimand is the unweighted mean over the pre-specified panel, and removing them after observing outcomes would change that estimand.

The operator also has a narrower construct than "hand evidence" broadly. Gaussian low-pass blur suppresses fine appearance detail while largely preserving coarse hand position and gross trajectory, so the contrast does not isolate handshape from all other fine manual appearance cues. Provider-side resizing and temporal preprocessing are partly unobservable and may attenuate or transform the effective perturbation before inference. A stronger follow-up would pre-specify a larger model panel, evaluate multiple signers, and use interventions that separately target fine handshape/orientation cues, coarse trajectory/location, and complementary non-manual evidence.

7 Conclusion

Across 150 isolated Chinese signs and a fixed four-model video-MLLM panel, the pre-specified C–T contrast is +3.17 percentage points (95% CI $[-0.17, +6.50]$; sign-flip $p = 0.08172$). The interval includes zero, so the result is directional rather than confirmatory; the four positive model-level point estimates are descriptive rather than separate confirmatory evidence. Matched visual interventions can complement benchmark accuracy by testing sensitivity to hypothesized cue locations while making the control's residual mismatches explicit.

8 Limitations

The study covers one signer, 150 isolated signs, one dataset, and a four-choice task, so it does not establish signer generalization, continuous signing, discourse, or open-ended translation. Dataset labels were not independently re-annotated, and fixed pre-outcome visual/geometric QC was not an independent ground-truth tracking study. The control does not match regional information content; provider resizing/frame sampling is partly unobservable; and no full-sample native-pixel σ distribution is asserted. Two Primary models are floor-limited. The fixed four-model panel does not support broad generalization across video MLLMs. The incomplete post-deblinding secondary branches do not enter Primary inference.

9 Ethical Considerations

We reuse NationalCSL-DP and recruit no new participants; the dataset paper reports informed consent and ethics-board approval for public release (Jin et al., 2025). The representative face in Figure 1 is pixelated only for publication; model inputs are unchanged. We frame errors as system limitations and encourage Deaf and sign-language user involvement in future accessibility studies.

Acknowledgments

This work was supported by the General Project of the Beijing Social Science Fund ("Research on the Intangible Cultural Heritage Attributes and

Living Transmission of Local Sign Languages: A Case Study of Beijing Sign Language," Grant No. 25BJ03128); the 2024 Key Project of the Sichuan Province Key Laboratory of Philosophy and Social Science for Language Intelligence in Special Education ("Processing Specifications for Phonological and Morphological Annotation of Sign Language: Corpus-Based Annotation Practice," Grant No. YYZN-2024-2); and the General Project of the National Language and Writing Promotion Base (Beijing Union University) ("Research on Automatic Multichannel Phonological Annotation Methods for Sign Language Corpora for Information Processing," Grant No. YW202504).

Generative AI tools were used to assist with code development, numerical cross-checking, and language and editorial refinement during manuscript preparation. All substantive research decisions, experimental design, analyses, interpretation of results, citation verification, and final manuscript content were reviewed and verified by the authors, who take full responsibility for the work.

References

Keren Artiaga, Sabyasachi Kamila, Haithem Afli, Conor Lynch, and Mohammed Hasanuzzaman. 2025. Rethinking sign language translation: The impact of signer dependence on model evaluation. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 18379–18391. Association for Computational Linguistics.

Anal Roy Chowdhury and Debarshi Kumar Sanyal. 2026. Can small vision–language models perform sign language translation? In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 32150–32166. Association for Computational Linguistics.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, and 2 others. 2025. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal LLMs in video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24108–24118.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, and 7 others. 2020. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1307–1323. Association for Computational Linguistics.

Wenyi Hong, Yean Cheng, Zhuoyi Yang, Weihan Wang, Lefan Wang, Xiaotao Gu, Shiyu Huang, Yuxiao Dong, and Jie Tang. 2025. MotionBench: Benchmarking and improving fine-grained video motion understanding for vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8450–8460.

Mert Inan, Anthony Sicilia, and Malihe Alikhani. 2025. SignAlignLM: Integrating multimodal sign language processing into large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3691–3706. Association for Computational Linguistics.

Tao Jiang, Xinchen Xie, and Yining Li. 2024. RTMW: Real-time multi-person 2D and 3D whole-body pose estimation. *Preprint*, arXiv:2407.08634.

Peng Jin, Hongkai Li, Jun Yang, Yazhou Ren, Yuhao Li, Lilan Zhou, Jin Liu, Mei Zhang, Xiaorong Pu, and Siyuan Jing. 2025. A large dataset covering the Chinese national sign language for dual-view isolated sign language recognition. *Scientific Data*, 12:660.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, Limin Wang, and Yu Qiao. 2024. MVBench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22195–22206.

Yuanxin Liu, Shicheng Li, Yi Liu, Yuxiang Wang, Shuhuai Ren, Lei Li, Sishuo Chen, Xu Sun, and Lu Hou. 2024. TempCompass: Do video LLMs really understand videos? In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 8731–8772. Association for Computational Linguistics.

Kayo Yin, Amit Moryossef, Julie Hochgesang, Yoav Goldberg, and Malihe Alikhani. 2021. Including signed languages in natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7347–7360. Association for Computational Linguistics.

Rui Zhao, Xuewen Zhong, Xiaoyun Zheng, Jinsong Su, and Yidong Chen. 2026. CNSL-bench: Benchmarking the sign language understanding capabilities of MLLMs on Chinese national sign language. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 40869–40890. Association for Computational Linguistics.

A Execution Details

Models and frozen request settings. Qwen: `qwen3.7-plus-2026-05-26`, Base64 `video_url`, exposed fps=10, temperature=0, max tokens=16, thinking disabled, stream=false. GLM: `glm-5v-turbo`, Base64 `video_url`, provider-native video preprocessing, temperature=0, max tokens=16, thinking disabled, stream=false. Doubao: `doubao-seed-2-0-pro-260215`, frozen Base64-video route, provider-native preprocessing, temperature=0, max tokens=16, thinking disabled, stream=false. ERNIE: `ernie-4.5-turbo-vl`, MP4/mp4v Base64 data URL, exposed fps=5, `use_audio=false`, temperature=0, max tokens=16, stream=false; no explicit thinking parameter was recovered. Parameters not explicitly set (including top-p and a decoding seed) were left to provider defaults.

Task construction. The source vocabulary had 6,707 rows; frozen eligibility left 6,026. Target sampling used NumPy Generator/PCG64 seed 20260818 (180 without replacement; 1–150 FORMAL, 151–180 RESERVE), correct-position balancing seed 2026081801, and distractor seed 2026081802. Distractors came from the same eligible pool with target exclusion and normalized-English-translation uniqueness constraints, not semantic nearest-neighbor selection. All 180 candidate glosses mapped uniquely to `Participant_01-front/CSY/front` and passed the media gate, so no replacement was used.

Exact model-visible prompt. The 150-record UTF-8 prompt manifest has SHA-256 `75aa014aea247476aad81f97097d79b1e9aa2ba823cd589996a8c58c099b1d10`. Its fixed instruction is:

> 你将看到一段国家通用手语的孤立词视频。请仅根据视频内容，从以下四个候选词中选择最符合该手语表达含义的一项。只输出一个大写字母A、B、C 或D，不要解释，不要输出任何其他内容。

Pinyin transliteration: Ni jiang kan dao yi duan Guojia Tongyong Shouyu de guli ci shipin. Qing jin genju shipin neirong, cong yixia si ge houxuan ci zhong xuanze zui fuhe gai shouyu biaoda hanyi de yi xiang. Zhi shuchu yi ge daxie zimu A, B, C huo D, bu yao jieshi, bu yao shuchu renhe qita neirong.

English translation: You will see a video of an isolated word in Chinese National Common Sign Language. Based only on the video content, choose from the following four candidate words the one that best matches the meaning expressed by the sign. Output only one uppercase letter A, B, C, or D. Do not explain and do not output anything else.

The four candidate glosses then follow as A–D. For a given item, prompt bytes and A/B/C/D order are identical across O/T/C and all Primary models.

Retries and parsing. Each scheduled call has at most three attempts: attempt 2 after 5 s and attempt 3 after an additional 20 s, only for transient transport/network/timeout/408/425/429/5xx failures. Retries keep the same model, video, prompt, and provider parameters. HTTP-200 empty/refusal/safety/malformed outputs are scientific outputs and are not retried for quality; permanent non-transient 4xx errors are infrastructure failures. Returned text is NFKC-normalized with whitespace removed; an exact A/B/C/D or unique leading answer is accepted, while conflicting/non-unique outputs are malformed scientific outputs. Final Primary malformed count is zero.

Author Index

Ahmed, Shoaib, 18
Ali, Md. Asif, 70, 77
Apurbo, Md. Mosharaf Hossain, 70, 77
Artiaga, Keren, 1, 11

Ballester, Coloma, 177
Barberà, Gemma, 54
Bhuyan, M.K., 213

Chandaliya, Praveen Kumar, 237
Chiruzzo, Luis, 111
Cortés, Joan Gomà, 54

Dattatreya, Pokala, 1, 11

Ebling, Sarah, 84
Elaswad, Ali, 224
Escolano, Carlos, 54

Farrarós, Carolina Del Corral, 177

Granero Moya, Marcel, 177
Gruber, Ivan, 1, 11

Haro, Gloria, 177
Hasan, Kazi Reyazul, 70, 77
Hasanuzzaman, Mohammed, 1, 11
Honzík, Jakub, 1, 11
Hrúz, Marek, 1, 11
Hsieh, Shu-Kai, 43

Ingimundarson, Finnur Ágúst, 84

Jin, Peng, 243
Jing, Siyuan, 243
Joshi, Abhinav, 1, 11

Kamila, Sabyasachi, 1, 11
Kapitanov, Alexander, 147, 193, 203
Korotaev, Pavel, 193, 203
Krishna, Pallapu Mohan, 213
Kumar, Uppala Sanjay, 213
Kvanchiani, Karina, 147, 193

Lanza, Matteo, 96
Lim, Chunki, 43
Lisboa, Emilio Valle, 111
Liu, Hongzhe, 243

Mahmud, Adibah, 172
Marques, Ricardo, 177
Mathias, Sandeep, 32
Melero, Maite, 54
Mishra, Dipti, 18
Mishra, Pruthwik, 237
Modi, Ashutosh, 1, 11
Müller, Mathias, 84

Nagaev, Alexander, 147, 193, 203
Nam, Nguyen Tu, 141
Neog, Debanga Raj, 213

Özel, Özge Can, 162

Peng, Kai-Yang, 43
Petrova, Elizaveta, 147
Poornima, Galiveeti, 32
Pujara, Khushi Jaydeep, 1, 11

Quiroga, Facundo Manuel, 1, 11

Rahman, M Saifur, 172
Rosá, Manu Bervejillo, 111

Saha, Kowshik, 172
Samanvitha, Bolisetty, 237
Sant, Gerard, 54
Saqr, Omar Mokhtar Gaber, 224
Saraclar, Murat, 162
Shrestha, Gyanendra, 121, 132
Singh, Sanjeet, 1, 11
Srivastava, Ojas, 237
Stassi, Ariel E., 111
Surovtsev, Petr, 147, 193, 203
Şahin, Saliha Nur, 162

Takahashi, Hiroki, 141
Techera, Facundo, 111

Xu, Cheng, 243

Yao, Dengfeng, 243
Ye, Bowen, 243

Zhang, Liting, 121, 132

Þorvaldsdóttir, Guðný Björk, 84

Železný, Tomáš, 1, 11