

SpaMo-BdSL: Gloss-Free Bangla Sign Language Translation Using Spatial Configurations and Motion Dynamics with LLMs

Kowshik Saha¹, Adibah Mahmud², and M Saifur Rahman³

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology

¹kowshiksahakabya@gmail.com, ²adibahmahima@gmail.com, ³mrahman@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 peo-

ple 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

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 $\mathcal{L} = \mathcal{L}_{CE} + \alpha \mathcal{L}_{CLIP}$ 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

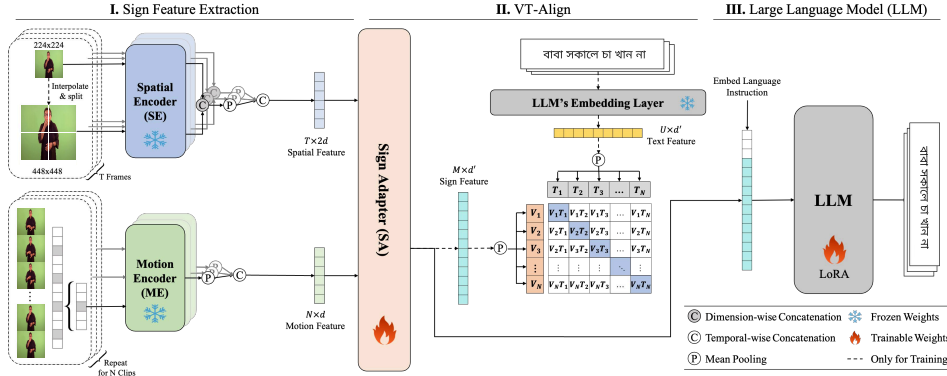


Figure 1: Overview of the SpaMo-BdSL pipeline. A frozen CLIP ViT-L/14 extracts spatial features and a frozen VideoMAE-L/16 extracts motion features; these are projected, fused by a temporal convolutional Sign Adapter, and passed with a Bangla prompt to a LoRA-tuned text model that generates the sentence. 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 (Bhat-

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.

tacharjee 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.

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

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.

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.

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, Yanzhi 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 pre-training](#). 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.