SAME: Signer-Aware Mixture-of-Experts for Test-Time Adaptation in Sign Language Translation

Lujia Yang, Weicai Yan, Yongbo He, Qifei Zhang,
Tao Jin, Jinshan Zhang*, Meng Xi, Jianwei Yin
Zhejiang University, Hangzhou, China
{yanglujia, yanweicai, cstzhangqf, jint_zju, zhangjinshan, ximeng}@zju.edu.cn
zjuyjw@cs.zju.edu.cn, reheybo@gmail.com

Abstract

Sign language translation (SLT) is essential for bridging communication between the deaf and hearing communities, but real-world deployment suffers from domain shift such as signer variability, lighting, and background changes. Supervised fine-tuning is impractical due to limited labeled data, and existing unsupervised adaptation methods require batch statistics or long adaptation. We introduce Test-Time Adaptation (TTA) for SLT, enabling rapid adaptation to domain shift without the need for labeled data. To the best of our knowledge, this is the first study to explore TTA in SLT. Existing TTA methods predominantly focus on image classification tasks and lack a comprehensive strategy for handling domain shift in SLT. In response, we introduce SAME, a plug-and-play, signer-aware Mixture-of-Experts (MoE) TTA architecture for SLT. SAME inserts lightweight MoE modules after multiple encoder layers. Gates are conditioned on signer features and stabilized with unsupervised regularizers, effectively decoupling domain shift across encoder depths while enabling personalized adaptation. Experiments show that SAME outperforms existing TTA methods and can enhance the capabilities of multiple SLT models.

1 Introduction

Sign language translation (SLT) has emerged as a crucial technology for bridging communication barriers between the deaf community and hearing society. However, real-world deployment faces domain shift such as signer variation, lighting changes, and background clutter, which degrades translation quality. While modern SLT models (Wong et al., 2024; Li et al., 2025) have demonstrated strong generalization in controlled settings, edge cases, such as atypical signer kinematics or recordings captured under novel conditions, can still lead to substantial performance drops, which are unacceptable in practical assistive contexts. Traditional approaches, such as post-hoc supervised fine-tuning on labeled target data, are impractical in real-time deployment due to the difficulty of obtaining labeled data. Unsupervised domain adaptation techniques, including batch-norm-based methods, often require long adaptation times and batch statistics that are unavailable in streaming settings or introduce instability. To this end, we introduce Test-Time Adaptation (TTA) for SLT, enabling unsupervised adaptation without target-domain labels during test time. The task of TTA for sign language translation is shown in Fig. 1.

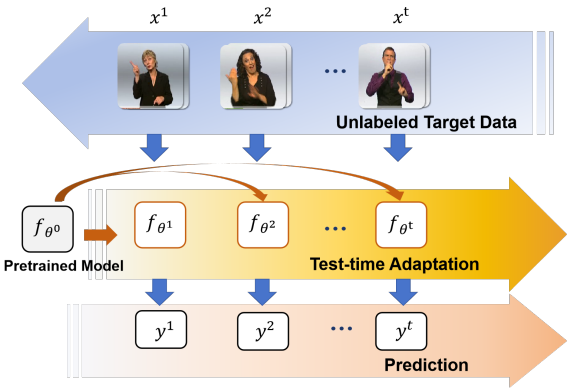


Figure 1: An illustration of our TTA for SLT. In this setting, unlabeled target samples arrive sequentially, and a pre-trained SLT model adapts at test time using unsupervised objectives, producing translation outputs in real time.

TTA aims to adapt pre-trained models to new domains using only unlabeled target data during inference. Existing TTA methods can be broadly categorized into optimization-based (Zhang et al., 2025; Zhou et al., 2025), data-based (Gong et al., 2023), and model-based approaches (Shin and Kim, 2024; Lee et al., 2024). In practice, TTA frameworks tend to combine multiple methods from different categories to make better adaptation. While

*Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 21274–21289

these methods have achieved promising results in several domains, the majority of existing work remains focused on image and speech areas, with limited exploration in video or sequence generation settings, let alone in the task of SLT.

To address these gaps, we introduce SAME, a signer-aware Mixture-of-Experts (MoE) TTA architecture for SLT. We augment the SLT encoders with lightweight SAME modules placed after each encoder layer for fine-grained, depth-wise adaptation. Each SAME contains several LoRA-based experts and a pass-through branch to balance expert usage without forcing adaptation. Signer features are fed into the gating network for personalized expert selection and better domain information capture. These designs decompose domain shift across feature depths and experts, allowing shallow and deep features to adapt in complementary ways. In order to enable experts to better decouple domain shift, we introduce a expert diversity loss, which, together with the task loss, is used to initialize SAME on small labeled source subset. During inference, a few gradient steps are applied to SAME using unsupervised TTA losses, including entropy minimization, minimum class confusion, and pseudo-label supervision, to enable stable adaptation.

We conduct extensive experiments on three SLT benchmarks: Phoenix-2014T (Camgoz et al., 2018), CSL-Daily (Zhou et al., 2021a) and How2Sign (Duarte et al., 2021). Experiments show that SAME consistently outperforms existing TTA approaches and enhances the adaptability of multiple SLT architectures under domain shift.

- We present the first study of TTA for SLT to the best of our knowledge, providing a practical solution for the challenging problem of unsupervised adaptation during test time.
- We propose a sparse signer-aware MoE framework with lightweight expert modules, signer-aware gating, designed initialization and unsupervised regularization for stable TTA.
- Experiments show that our method outperforms existing TTA methods and generalizes across multiple SLT models.

2 Related Work

2.1 Sign Language Translation

SLT aims to convert sign language videos into spoken language text and can be categorized into gloss-based and gloss-free paradigms. Glosses are the word-for-word transcription of sign language. Gloss-based approaches, such as SLRT (Camgoz et al., 2020) and STMC-T (Zhou et al., 2021b), leverage intermediate gloss supervision with CTC or alignment losses, while unified multi-task frameworks like TS-SLT (Chen et al., 2022b) and SLTUNet (Zhang et al., 2023) integrate gloss recognition and translation for better shared representations. To reduce reliance on costly gloss annotations, works (Zhou et al., 2023; Wong et al., 2024; Gong et al., 2024; Li et al., 2025) have advanced gloss-free SLT and explored the combination with large language models. Despite these advances, most models assume fixed test distributions and lack mechanisms for unsupervised adaptation.

2.2 Test-Time Adaptation

Test-Time Adaptation aims to adapt pre-trained models to domain shift using only unlabeled target data during inference. Existing TTA methods can be broadly categorized into optimization-based, data-based, and model-based approaches (Wang et al., 2025). Optimization-based methods are the most commonly used ones, including normalization calibration (Wang et al., 2021; Zhou et al., 2025), teacher–student optimization (Yuan et al., 2023), and specially designed optimization objectives (Niu et al., 2023, 2022; Zhang et al., 2025; He et al., 2026). Data-based methods (Gong et al., 2023) enhance robustness through strategies such as data augmentation and leverage prediction consistency to improve model generalization. Model-based approaches (Shin and Kim, 2024; Lee et al., 2024) instead adjust the model architecture, for instance by adding or replacing modules, to facilitate adaptation. In practice, TTA frameworks usually combine multiple methods. TTA has seen significant success in many domains, but has not been extensively explored for sequence generation tasks like SLT.

2.3 Mixture-of-Experts

Mixture-of-Experts is initially proposed as an architecture for efficiently scaling model capabilities (Jacobs et al., 1991; Jordan and Jacobs, 1994), consisting of multiple expert networks and a gating module that dynamically assigns weights to experts and fuses their outputs (Dimitri et al., 2025). It has been widely adopted in natural language processing (NLP) (Rajbhandari et al., 2022; Xue et al., 2024) and computer vision (Videau et al., 2024;

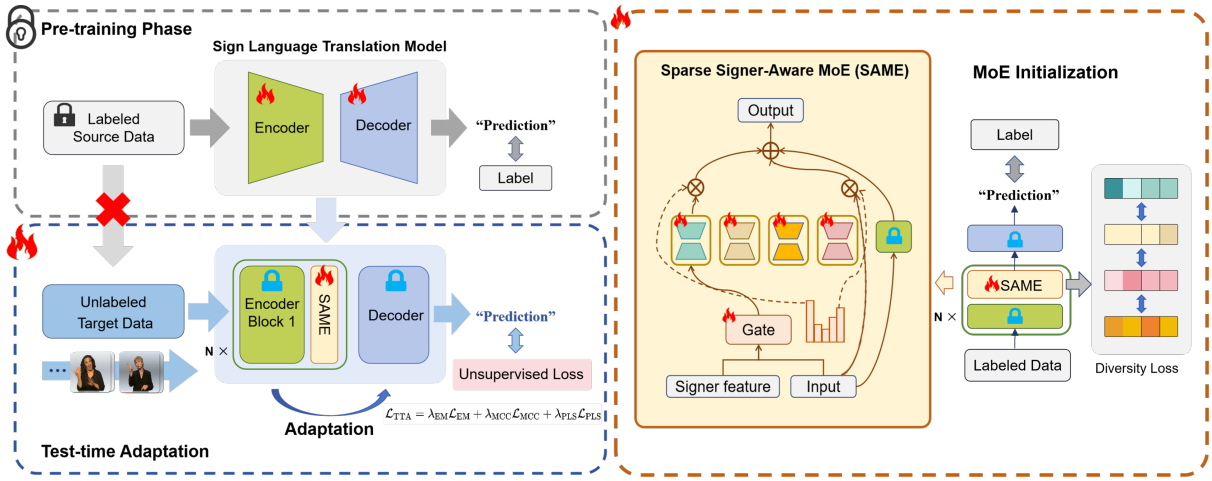


Figure 2: The framework of the proposed method SAME. The SLT model is pre-trained with labeled data. At test time, the pre-trained parameters are frozen, and Signer-Aware Mixture-of-Experts (SAME) modules are inserted after each encoder layers. SAME consists of multiple experts and a pass-through branch, with a signer-aware gating network to guide personalized expert selection. The modules are adapted with unlabeled target sample under designed unsupervised objectives. To stabilize adaptation, SAME is initialized on a small set of labeled data using both SLT task loss and the proposed diversity loss.

Rossi et al., 2025). To reduce the computational overhead of MoE, recent works (Zhu et al., 2024; Chen et al., 2024) have explored lightweight variants by integrating parameter-efficient tuning techniques such as LoRA (Hu et al., 2022), enabling scalable expert deployment without incurring prohibitive memory costs. However, MoE has been rarely explored in sign language translation, particularly under the test-time adaptation setting where domain shift emerges dynamically.

3 Preliminary

Sign Language Translation. The goal of SLT is to translate a video sequence of sign language into a spoken language sentence. Most SLT tasks can be regarded as encoder-decoder architectures. Given an input video $X = (x_1, \ldots, x_T)$ with T frames, an encoder $\mathbf{E}$ extracts sequential features. A decoder $\mathbf{D}$ then autoregressively generates the target sentence $\hat{Y} = (\hat{y}_1, \ldots, \hat{y}_N)$:

$$\hat{Y} = \mathbf{D}(\mathbf{E}(X)), \tag{1}$$

SLT can follow a gloss-based paradigm, leveraging intermediate gloss supervision, or a gloss-free paradigm, directly performing video-to-text translation. In traditional SLT training, a model is optimized with cross-entropy loss on labeled source data.

Test-Time Adaptation. In this work, we focus on addressing the challenge of adapting a pre-trained model to unlabeled target data under domain shift without access to source data during adaptation. Let $f_{\theta_{\text{src}}}$ be a source model trained on labeled source data $\mathcal{D}_{\text{src}} = \{(X_{\text{src}}, Y_{\text{src}})\} \sim p_{\text{src}}(X, Y)$. At test time, the model encounters unlabeled target samples $\{X^i_{\text{tgt}}\}_{i=1}^n \sim p_{\text{tgt}}(X)$, where $p_{\text{tgt}}(X) \neq p_{\text{src}}(X)$ due to domain shift. In our TTA setting, samples arrive sequentially. The model is reset to θ_{src} for each instance and updated with small adaptation steps using only current sample X^i_{tgt}. Since target labels are unavailable, optimization relies on an unsupervised adaptation loss $\mathcal{L}_{\text{TTA}}$. The adapted model θ^i_{tgt} and the prediction $\hat{Y}^i_{\text{tgt}}$ are obtained as:

$$\theta^i_{\text{tgt}} = \text{Adapt}(\theta_{\text{src}}, X^i_{\text{tgt}}), \quad \hat{Y}^i_{\text{tgt}} = f_{\theta^i_{\text{tgt}}}(X^i_{\text{tgt}}), \tag{2}$$

where Adapt$(\cdot)$ defines the adaptation rule.

4 Method

We propose a signer-aware MoE-based TTA strategy for SLT. Our design introduces lightweight signer-aware MoE modules for depth-wise domain shift decoupling and personalized expert selection (Sec. 4.1), supervised MoE initialization with expert diversity loss to stabilize expert behavior (Sec. 4.2), and unsupervised TTA objectives for robust adaptation (Sec. 4.3). An overview of the proposed method is presented in Fig. 2.

4.1 Sparse Signer-Aware Mixture-of-Experts

To address signer-induced domain shift, we design a sparse signer-aware MoE module that combines parameter-efficient low-rank experts with sparse signer-aware routing to enhance the model's capacity while controlling the computational cost.

4.1.1 Low-Rank Experts

Each expert in our SAME module is parameterized as a low-rank adapter for parameter efficiency. Given an encoder feature sequence $h \in \mathbb{R}^{T\times d}$, the j-th expert $\mathbf{e}_j$ is applied to each temporal token independently as:

$$\mathbf{e}_j(h_t) = \frac{\alpha}{r} B_j A_j h_t, \quad t = 1, \ldots, T, \tag{3}$$

where $B_j \in \mathbb{R}^{d\times r}$ and $A_j \in \mathbb{R}^{r\times d}$ are trainable parameters, $r \ll d$ is the rank, and α/r is a scaling factor for stable optimization.

4.1.2 Sparse Signer-Aware Routing

To better capture domain variations, we enhance the gating mechanism with signer features s extracted via a pretrained MLP to encode signer-specific traits. These features are utilized by the gating network to guide expert selection. The gating function is implemented as a linear layer followed by a softmax activation, taking both modality features h and signer features s as input. We adopt a sparse MoE architecture coupled with a top-k routing policy:

$$\begin{aligned} &z(h_t, s) = W_g[h_t; s] + b_g, \\ &G(h_t, s) = \text{softmax}(\text{TopK}(z(h_t, s) + \mathcal{R}_{\text{noise}}, k)). \end{aligned} \tag{4}$$

Here, $G(\cdot)$ denotes the gating weights, $z(h_t, s)$ denotes the routing logits, $[h_t; s]$ denotes concatenation, $\mathcal{R}_{\text{noise}}$ represents the noise to encourage expert exploration, W_g and b_g are the parameters of the gating network. The $\text{TopK}(\cdot, k)$ operator selects the top-k routing entries by gating scores and masks the rest to $-\infty$ before softmax. We adopt frame-wise signer-aware routing, allowing different temporal positions to activate different experts, enabling experts to specialize in fine-grained temporal variations (e.g., gloss transitions and tempo changes), while adaptation remains sample-level.

4.1.3 MoE for Depth-wise Domain Decoupling

We insert SAME modules after each encoder layer of the SLT model. Each SAME comprises multiple experts and a pass-through branch, enabling both adaptive specialization and stable routing. Let $H_l = [h_{l,1}, \ldots, h_{l,T_l}] \in \mathbb{R}^{T_l\times d}$ denote the output of the l-th encoder layer, where $h_{l,t}$ is the t-th temporal token. The adapted representation is computed as:

$$\begin{aligned} &\text{SAME}(h_{l,t}, s) = \sum_{j=1}^{M} G_j(h_{l,t}, s)\, \mathbf{e}_j(h_{l,t}), \\ &\tilde{h}_{l,t} = h_{l,t} + \text{SAME}(h_{l,t}, s), \quad l = 1, \ldots, L, \end{aligned} \tag{5}$$

where M is the number of experts, L is the number of encoder layers, $G_j(h_{l,t}, s)$ is the routing weight of the j-th expert, and the router selects from the M experts together with a pass-through branch. When the pass-through branch is selected, $\text{SAME}(h_{l,t}, s) = 0$, so that $\tilde{h}_{l,t} = h_{l,t}$. The adapted output $\tilde{H}_l = [\tilde{h}_{l,1}, \ldots, \tilde{h}_{l,T_l}]$ is then fed into the next encoder layer.

4.2 MoE Initialization with Diversity Loss

Since random initialization of MoE may lead to expert collapse, where the router tends to favor the same expert, we initialize SAME on a small labeled set using the standard cross-entropy SLT loss together with an expert diversity regularizer. Given expert outputs $E \in \mathbb{R}^{B\times M\times T\times D}$ where B is batch size, M is the number of experts, T is the sequence length, and D is the hidden dimension, we flatten each output and encourage pairwise orthogonality among experts:

$$\mathcal{L}_{\text{div}} = \frac{1}{B}\sum_{b=1}^{B} ||(O_b \cdot O_b^T)/(T \cdot D) - I||_F^2, \tag{6}$$

where $O_b \in \mathbb{R}^{M\times(T\cdot D)}$ is the flattened expert output for sample b, and I is the identity matrix. The MoE initialization loss is:

$$\mathcal{L}_{\text{init}} = \mathcal{L}_{\text{SLT}} + \lambda_{\text{div}}\mathcal{L}_{\text{div}}. \tag{7}$$

This helps experts develop complementary behaviors, avoiding mode collapse and improving downstream adaptation.

4.3 Unsupervised Adaptation Objectives for Stable TTA

To ensure stable test-time adaptation with unlabeled target data, we adopt three unsupervised objectives: Entropy Minimization (EM), Minimum Class Confusion (MCC), and Pseudo-Label Supervision (PLS) inspired by prior TTA works in ASR (Lin et al., 2022, 2024). Let $P_i = p_\theta(y_i \mid$

$y_{<i}, x) \in \mathbb{R}^C$ denote the predicted distribution at the i-th decoding step for a generated sequence of length N. The objectives are formulated as:

$$\mathcal{L}_{\text{EM}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} P_{ij}\log P_{ij},$$
$$\mathcal{L}_{\text{MCC}} = \sum_{j=1}^{C}\sum_{j'\neq j} P_{\cdot j}^{\top} P_{\cdot j'}, \tag{8}$$
$$\mathcal{L}_{\text{PLS}} = \frac{1}{N}\sum_{i=1}^{N} \text{CE}(\hat{y}_i^{\text{src}}, P_i).$$

Here, EM encourages confident predictions, MCC reduces inter-class confusion by promoting clearer class separation, and PLS acts as a soft constraint that prevents the adapting model from drifting too far from the source model by leveraging pseudo labels $\hat{y}_i^{\text{src}}$ from the frozen source model. Their combination improves prediction consistency and robustness during test-time adaptation. The overall adaptation objective is a weighted sum:

$$\mathcal{L}_{\text{TTA}} = \lambda_{\text{EM}}\mathcal{L}_{\text{EM}} + \lambda_{\text{MCC}}\mathcal{L}_{\text{MCC}} + \lambda_{\text{PLS}}\mathcal{L}_{\text{PLS}}. \tag{9}$$

5 Experiments

We introduce the experimental setup in Sec. 5.1, followed by comparisons with multiple TTA baselines and different SLT models in Sec. 5.2. Sec. 5.3 explores the effect of each component of our framework. More analysis can be found in Sec. 6.

5.1 Experimental Settings

5.1.1 Datasets and Evaluation metrics

We evaluate our approach on three SLT benchmark datasets: Phoenix-2014T, CSL-Daily, and How2Sign. Phoenix-2014T comprises continuous German Sign Language videos paired with spoken language transcripts and gloss annotations. CSL-Daily provides large-scale daily conversational Chinese Sign Language data. How2Sign consists of American Sign Language videos annotated with English sentences. Treating signer variation as domain shift, we pre-train the SLT model on a signer subset of each dataset with full supervision, then adapt it to unseen signers under the TTA setting. Performance is quantified via BLEU-1 to BLEU-4 (Papineni et al., 2002), with final results reported as the average across all unseen signers.

5.1.2 Baselines

Since existing SLT works involve heterogeneous backbones and additional modules which introduce confounding factors and hinder clear attribution of improvements to TTA methods, we use a lightweight transformer-based SLT model as the source and compare our approach with representative TTA methods on it for fair comparison, including optimization-based methods TENT (Wang et al., 2021), EATA (Niu et al., 2022), SAR (Niu et al., 2023) and AEO (Dong et al., 2025), and model-based approaches CoTTA (Wang et al., 2022) and BeCoTTA (Lee et al., 2024). In addition, we further apply SAME to several existing SLT frameworks, including the gloss-based methods SLRT (Camgoz et al., 2020), MMTLB (Chen et al., 2022a), TS-SLT (Chen et al., 2022b), and the gloss-free methods GFSLT-VLP (Zhou et al., 2023), Sign2GPT (Wong et al., 2024), GloFE-VN (Lin et al., 2023), SLT-IV (Tarrés et al., 2023), Uni-Sign (Li et al., 2025), to assess the generalizability and plug-in adaptability of our method across diverse architectures. For fair comparison, all SLT backbones are re-trained and evaluated under the same source/target signer split.

5.1.3 Implementation Details

For SAME, we set the number of experts to 4 and use LoRA modules with rank 8 as the expert networks. We optimize using AdamW with a learning rate of 1×10^{-3}, performing 5 adaptation steps per test sample. Loss weights are set as $\lambda_{\text{div}} = 0.5$, $\lambda_{\text{PLS}} = 0.2$, $\lambda_{\text{EM}} = 0.3$ and $\lambda_{\text{MCC}} = 0.7$. All TTA experiments are conducted on a single NVIDIA RTX 4090 GPU. All main results are averaged over three runs with different random seeds, while ablation results are obtained from a single run.

5.2 Main Results

5.2.1 Comparison with TTA methods

We compare our method with representative TTA approaches on the Phoenix-2014T and CSL-Daily datasets, where all methods are adapted on the same source model under the TTA setting. Results in Tab. 1 show that SAME achieves the best performance across all metrics and datasets, outperforming both optimization-based and model-based TTA baselines and gaining a BLEU-4 improvement of +2.73 on Phoenix-2014T and +2.21 on CSL-Daily. Optimization-based methods such as TENT, SAR and EATA lag significantly in performance, while model-based approaches (CoTTA, BeCoTTA) deliver stronger results via auxiliary structures yet still fall short of SAME.

Method	Phoenix-2014T					CSL-Daily				
	B1	B2	B3	B4	ΔB4	B1	B2	B3	B4	ΔB4
Source	31.06	19.00	13.25	10.02	/	30.43	16.94	9.91	5.88	/
TENT	32.78	19.41	13.69	10.63±0.16	+0.61	30.49	16.30	9.10	5.92±0.26	0.04
EATA	29.76	17.38	11.95	9.43±0.40	-0.59	29.12	16.08	9.30	5.78±0.31	-0.1
CoTTA	34.40	20.97	14.76	11.35±0.36	+1.33	27.18	16.04	9.50	5.99±0.29	+0.11
SAR	35.48	21.75	15.39	11.83±0.20	+1.81	30.43	17.49	10.83	6.90±0.24	+1.02
BeCoTTA	36.26	22.49	15.88	12.22±0.18	+2.20	30.04	17.01	10.53	6.70±0.20	+0.82
AEO	30.49	17.28	11.60	8.58±0.43	-1.44	24.75	12.59	7.41	4.54±0.35	-1.34
SAME(ours)	**37.68**	**23.94**	**16.39**	**12.75±0.15**	**+2.73**	**33.76**	**20.59**	**12.81**	**8.09±0.18**	**+2.21**

Table 1: Comparative results on Phoenix-2014T and CSL-Daily. (Bold: best results, B: BLEU. Results are averaged over 3 runs.)

Method	Phoenix-2014T					CSL-Daily				
	B1	B2	B3	B4	ΔB4	B1	B2	B3	B4	ΔB4
SLRT	30.92	18.17	12.57	9.53	/	21.44	10.18	6.73	4.88	/
MMTLB	40.54	29.93	21.31	17.17	/	37.58	24.86	16.08	13.62	/
TS-SLT	42.03	30.41	22.91	18.23	/	40.15	27.24	20.95	15.25	/
GFSLT-VLP	36.09	21.71	15.32	12.13	/	25.60	13.72	8.21	5.02	/
Sign2GPT	36.48	22.57	15.67	12.48	/	30.45	18.05	11.16	7.10	/
SLRT+SAME	36.65	21.37	15.02	11.53±0.20	+2.00	31.58	19.01	11.79	7.67±0.20	+2.79
MMTLB+SAME	41.69	31.34	22.13	18.97±0.14	+1.80	39.11	25.47	18.40	14.37±0.15	+0.75
TS-SLT+SAME	44.33	32.22	25.21	19.97±0.15	+1.74	43.24	29.81	22.79	16.82±0.14	+1.57
GFSLT-VLP+SAME	38.18	25.87	19.10	15.52±0.22	+3.39	30.81	17.43	10.73	6.88±0.21	+1.86
Sign2GPT+SAME	37.51	23.42	16.89	13.87±0.16	+1.39	34.73	20.25	12.99	9.53±0.17	+2.43

Table 2: Effect of SAME on SLT models across Phoenix-2014T and CSL-Daily. (B: BLEU. Results with SAME are averaged over 3 runs.)

5.2.2 Results on Existing SLT Works

To verify the generalizability of our framework, Tab. 2 and Tab. 3 report the results of integrating our SAME module into various existing SLT architectures. All evaluated SLT models, including gloss-based and gloss-free models, exhibit consistent BLEU improvements with positive ΔB4 values after SAME integration across the three datasets. These cross-dataset and cross-model gains confirm that SAME is not only effective for simplified source models but also robustly generalizes to more complex SLT systems and datasets alike.

5.3 Ablation Study

We conduct ablation experiments on Phoenix-2014T to analyze the effect of adaptation targets, components of SAME and TTA losses as well as their hyperparameters.

5.3.1 Where to Adapt

To explore which parts of the model are most effective to adapt during TTA, we compare diverse adaptation targets: full-parameter tuning, normalization-layer tuning, LoRA-only, MoE-only, and the hybrid MoE+LoRA (our method) adaptation. The performance and number of trainable parameters of each target are shown in Tab. 4. As shown, lightweight modules (LoRA, MoE) consistently outperform normalization-layer tuning, while full-parameter tuning yields better results but incurs slower convergence and higher computational overhead. SAME achieves the best performance with merely 0.63% trainable parameters, striking an effective balance between adaptation capability and efficiency.

5.3.2 Ablation on SAME Components

We analyze the contribution of each component and report the results in Tab. 5. We observe that ex-

Method	B1	B2	B3	B4	ΔB4
GloFE-VN	8.75	4.62	1.26	0.56	/
SLT-IV	25.66	11.57	6.02	3.38	/
Uni-Sign	33.96	19.22	13.67	9.19	/
GloFE-VN+SAME	10.08	5.04	1.83	0.92±0.11	+0.36
SLT-IV+SAME	31.68	16.27	9.21	5.44±0.18	+2.06
Uni-Sign+SAME	35.32	21.92	15.11	10.24±0.16	+1.05

Table 3: Effect of SAME on SLT models on How2Sign. (B: BLEU. Results with SAME are averaged over 3 runs.)

Adaptation Target	B1	B2	B3	B4	ΔB4	Param	Param%
Full Parameters	35.62	22.77	15.74	12.62	+2.60	35.99M	100.00%
Norm Layers	33.10	20.05	14.22	11.38	+1.36	28.67K	0.79%
LoRA	36.66	21.46	15.24	12.06	+2.04	73.73K	0.20%
MoE	36.54	22.19	15.72	12.55	+2.53	6.33M	17.60%
MoE + LoRA (Ours)	**37.61**	**23.94**	**16.43**	**12.77**	**+2.75**	227.36K	0.63%

Table 4: Effect of adaptation targets. (Bold: best results, B: BLEU.)

Method	B1	B4
Full SAME	**37.61**	**12.77**
w/o Signer-Aware Routing	31.53	11.91
w/o Frame-wise Expert	36.87	12.17
w/o Pass-through	36.98	12.32
w/o MoE Initialization	33.31	12.01

Table 5: Effect of each component of SAME. (Bold: best results, B: BLEU.)

cluding any single module leads to a performance degradation, validating the necessity of each design. Among them, removing the signer-aware routing causes the most significant drop, suggesting the importance of personalized expert selection. Eliminating MoE initialization also results in substantial decline, indicating that well-initialized experts help stabilize TTA. The frame-wise expert selection and pass-through branch yield moderate but consistent improvements, reflecting their roles in decomposing domain shift into finer temporal segments and achieving more balanced adaptation, respectively.

5.3.3 Ablation on Unsupervised TTA Losses

We investigate the impact of different unsupervised TTA losses, including entropy minimization (EM), minimum class confusion (MCC), and pseudo-label supervision (PLS). As shown in Tab. 6, using a single objective yields limited gains, while combining EM, MCC, and PLS consistently improves performance. Removing any component degrades results, demonstrating their complementary effects.

EM	MCC	PLS	B1	B2	B3	B4
✓	-	-	34.60	19.76	14.42	11.93
-	✓	-	34.04	20.53	14.17	11.55
-	-	✓	31.65	18.69	12.98	10.84
✓	✓	-	37.45	21.55	16.22	12.54
-	✓	✓	35.60	20.76	14.42	11.93
✓	-	✓	36.06	22.55	15.64	12.40
✓	✓	✓	**37.61**	**23.94**	**16.43**	**12.77**

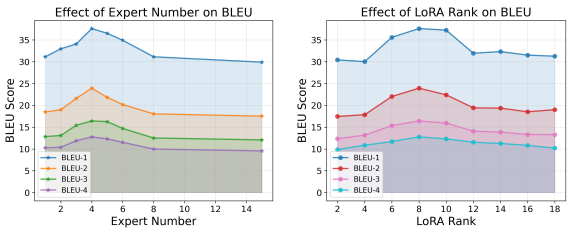
Table 6: Effect of each proposed TTA loss. (Bold: best results, B: BLEU.)

Top-k	B1	B2	B3	B4
1	**37.61**	**23.94**	**16.43**	**12.77**
2	32.98	19.94	14.34	11.68
3	31.15	19.06	13.75	11.21
4	32.2	19.65	13.97	11.18

Table 7: Effect of Top-k. (Bold: best results, B: BLEU.)

5.3.4 Effect of the Hyperparameters of Experts

Fig. 3 shows how expert numbers, top-k routing, and the LoRA rank influence model performance. As shown in Fig. 3 (a), increasing expert number initially improves BLEU scores, peaking at 4 ex-



(a) Effect of Expert Number on BLEU (b) Effect of LoRA Rank on BLEU

Figure 3: Effect of expert number (a) and LoRA rank (b) on model performance.

perts, after which performance gradually declines due to over-fragmentation and weakened specialization. For LoRA rank (Fig. 3 (b)), moderate values (rank = 8) achieve the best results, balancing parameter capacity and adaptation stability. Extremely low or high ranks degrade performance, suggesting either underfitting or redundancy. Similarly, results in Tab. 7 show that using top-1 routing consistently yields the best performance, indicating that sparse expert activation fosters stable adaptation.

6 In-depth Analysis

6.1 Analysis of Computational Cost

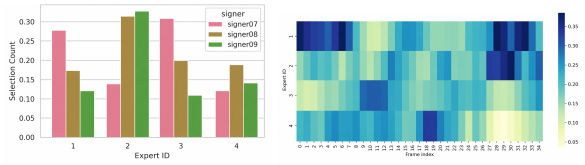
We compare the computational cost of SAME with prior TTA methods on Phoenix-2014T using the same transformer-based SLT backbone. Results in Tab. 8 show that SAME achieves a favorable balance between adaptation efficiency and performance improvement, making it suitable for real-world deployment scenarios.

Method	Latency (s/sample)	GFLOPs
Source	0.59	5.46
TENT	1.35	50.81
SAR	1.92	79.78
BeCoTTA	2.75	110.13
SAME (ours)	1.62	64.47

Table 8: Computational cost comparison of different TTA methods.

6.2 Analysis of Expert Behaviors

To analyze the role of signer-aware and frame-wise routing, we visualize expert usage across signers and over time on Phoenix-2014T (Fig. 4). As shown in Fig. 4 (a), different signers activate distinct subsets of experts, indicating that the gating mechanism captures signer-specific characteristics. Fig. 4 (b) shows that expert selection varies across frames within a video, with different experts activated at different temporal segments. This behavior aligns with the compositional nature of sign language, where a sentence consists of multiple sequential glosses with distinct motion patterns. By routing experts at the frame level, SAME can selectively activate experts specialized in different motions and gloss-related visual cues, thereby improving translation performance.



(a) Expert usage of different signers (b) Frame-wise gating maps

Figure 4: In-depth analysis of expert behaviors

7 Conclusion

In this work, we introduced SAME, a signer-aware MoE framework for TTA in SLT. To the best of our knowledge, this is the first study to solve this challenging but practical task. SAME decomposes domain shift across encoder depths and expert pathways, guided by signer-aware gating and regularized by unsupervised adaptation objectives. Experiments demonstrate our method outperforms existing TTA methods and achieves consistent improvements on multiple SLT models, offering a practical solution for real-world deployment.

Beyond SLT, the core idea of SAME is not limited to sign language translation. Since SAME performs test-time adaptation via conditional expert routing, it may be applicable to other tasks that involve style-specific variation, without requiring full retraining of the backbone. Moreover, SAME is naturally compatible with Transformer-based LLM architectures in a plug-and-play manner, where lightweight experts can be inserted into frozen layers. In this setting, conditioning the router on external context, rather than only token content, may offer a parameter-efficient mechanism for personalization and context-aware adaptation. We will further explore these extensions in future work.

Limitations

While SAME advances TTA for SLT, several limitations remain. Firstly, the adaptation relies on updates during inference, which inevitably introduce slight delays in real-time deployment. As with most TTA methods, the effectiveness of unsupervised adaptation is strongly dependent on the quality of streaming inputs, and remains vulnerable to severe noise.

Ethics Discussion

Our work aims to improve the robustness of sign language translation models under domain shift through test-time adaptation. While such adaptive systems can reduce bias and improve accessibility for deaf communities, potential risks include overfitting to particular signer styles or misinterpretation in real-world deployment. We recommend human-in-the-loop validation when integrating such systems into assistive applications.

Acknowledgments

Jinshan Zhang is the corresponding author. This work was supported by the Key Research and Development Jianbing Program of Zhejiang Province (No. 2023C01002), Hangzhou Major Project and Development Program (No. 2022AIZD0140), Yongjiang Talent Introduction Program (No. 2022A-236-G) and the Zhejiang Key Laboratory Project (2024E10001) .

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)*, 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.

Shaoxiang Chen, Zequn Jie, and Lin Ma. 2024. LLaVA-MoLE: Sparse mixture of LoRA experts for mitigating data conflicts in instruction finetuning MLLMs. *arXiv preprint arXiv:2401.16160*.

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.

Vasily Dimitri, Barbara Regina, and Magdolna Alfonz. 2025. A survey on mixture of experts: Advancements, challenges, and future directions. *Authorea Preprints*.

Hao Dong, Eleni Chatzi, and Olga Fink. 2025. Towards robust multimodal open-set test-time adaptation via adaptive entropy-aware optimization. In *International Conference on Learning Representations*.

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)*, pages 2735–2744.

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.

Taesik Gong, Yewon Kim, Taeckyung Lee, Sorn Chottananurak, and Sung-Ju Lee. 2023. SoTTA: Robust test-time adaptation on noisy data streams. *Advances in Neural Information Processing Systems*, 36:14070–14093.

Yongbo He, Zirun Guo, and Tao Jin. 2026. Decoupling stability and plasticity for multi-modal test-time adaptation. *arXiv preprint arXiv:2603.00574*.

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 *International Conference on Learning Representations*.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. 1991. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87.

Michael I. Jordan and Robert A. Jacobs. 1994. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6(2):181–214.

Daeun Lee, Jaehong Yoon, and Sung Ju Hwang. 2024. BECoTTA: Input-dependent online blending of experts for continual test-time adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 27072–27093. PMLR.

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*.

Guan-Ting Lin, Wei Ping Huang, and Hung-yi Lee. 2024. Continual test-time adaptation for end-to-end speech recognition on noisy speech. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 20003–20015, Miami, Florida, USA.

Guan-Ting Lin, Shang-Wen Li, and Hung-yi Lee. 2022. Listen, adapt, better WER: Source-free single-utterance test-time adaptation for automatic speech recognition. In *Proc. Interspeech 2022*, pages 2198–2202.

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.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. 2022. Efficient test-time model adaptation without forgetting. In *International conference on machine learning*, pages 16888–16905. PMLR.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. 2023. Towards stable test-time adaptation in dynamic wild world. In *International Conference on Learning Representations*.

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.

Samyam Rajbhandari, Conglong Li, Zhewei Yao, Minjia Zhang, Reza Yazdani Aminabadi, Ammar Ahmad Awan, Jeff Rasley, and Yuxiong He. 2022. DeepSpeed-MoE: Advancing mixture-of-experts inference and training to power next-generation AI scale. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 18332–18346. PMLR.

Leonardo Rossi, Vittorio Bernuzzi, Tomaso Fontanini, Massimo Bertozzi, and Andrea Prati. 2025. Swin2-MoSE: A new single image supersolution model for remote sensing. *IET Image Processing*, 19(1):e13303.

Jin Shin and Hyun Kim. 2024. L-TTA: Lightweight test-time adaptation using a versatile stem layer. In *Advances in Neural Information Processing Systems*, volume 37, pages 39325–39349.

Laia Tarrés, Gerard I. Gállego, Amanda Duarte, Jordi Torres, and Xavier Giró-i Nieto. 2023. Sign language translation from instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 5625–5635.

Mathurin Videau, Alessandro Leite, Marc Schoenauer, and Olivier Teytaud. 2024. Mixture of experts in image classification: What's the sweet spot? *arXiv preprint arXiv:2411.18322*.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. 2021. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*.

Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. 2022. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7201–7211.

Zixin Wang, Yadan Luo, Liang Zheng, Zhuoxiao Chen, Sen Wang, and Zi Huang. 2025. In search of lost online test-time adaptation: A survey. *International Journal of Computer Vision*, 133(3):1106–1139.

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*.

Fuzhao Xue, Zian Zheng, Yao Fu, Jinjie Ni, Zangwei Zheng, Wangchunshu Zhou, and Yang You. 2024. OpenMoE: An early effort on open mixture-of-experts language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 55625–55655. PMLR.

Longhui Yuan, Binhui Xie, and Shuang Li. 2023. Robust test-time adaptation in dynamic scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15922–15932.

Biao Zhang, Mathias Müller, and Rico Sennrich. 2023. SLTUNET: A simple unified model for sign language translation. In *International Conference on Learning Representations*.

Qingyang Zhang, Yatao Bian, Xinke Kong, Peilin Zhao, and Changqing Zhang. 2025. COME: Test-time adaption by conservatively minimizing entropy. In *International Conference on Learning Representations*.

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. 2021a. 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. 2021b. Spatial-temporal multi-cue network for sign language recognition and translation. *IEEE Transactions on Multimedia*, 24:768–779.

Xingzhi Zhou, Boyang Zhang, Zhiliang Tian, Yibo Zhang, Xin Niu, Ka Chun Cheung, Simon See, and Nevin L. Zhang. 2025. Resilient test-time adaptation by mitigating batch-normalization overfitting. In *ICASSP 2025 – 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Chenyang Zhu, Bin Xiao, Lin Shi, Shoukun Xu, and Xu Zheng. 2024. Customize Segment Anything Model for multi-modal semantic segmentation with mixture of LoRA experts. *arXiv preprint arXiv:2412.04220*.

A Transformer-based SLT Baseline

To maintain a fair and controllable comparison with representative TTA methods, we construct a lightweight transformer-based SLT model as the source model. This choice is motivated by three factors: (1) existing SLT models differ substantially in backbone architecture, making it difficult to establish a uniform comparison; (2) many methods introduce extra modules or carefully designed auxiliary losses, which may confound the effect of TTA itself; and (3) a simplified transformer-based model provides higher efficiency and better reproducibility.

Our model consists of four main parts: transformer-based video encoder $\mathbf{E}_V$, keypoint encoder $\mathbf{E}_K$, fusion encoder $\mathbf{E}_F$ and text decoder $\mathbf{D}$. When the model receives the input sign language video X, we follow the previous work (Chen et al., 2022b) to extract the video features V and keypoint features K from it. These features are then fed into the corresponding encoders to obtain modality-specific embeddings. The fusion encoder aggregates multimodal representations through a lightweight fully connected layer, producing a unified latent representation. The decoder then generates the textual translation sequence in an autoregressive manner, where each token is predicted based on previously generated tokens and the fused encoder states. The overall formula is as follows:

$$\hat{Y} = \mathbf{D}(\mathbf{E}_F([\mathbf{E}_V(V); \mathbf{E}_K(K)])). \quad (10)$$

To enable effective test-time adaptation, we integrate the proposed SAME module into each Transformer encoder layer. During adaptation, all parameters of the source SLT model are frozen, while only the parameters within the SAME modules are updated. The architecture of the model is shown in the Fig. 5.

B Signer Feature Extraction

To represent signer-specific characteristics, we employ a lightweight MLP pretrained on source-domain videos. The MLP takes visual features as input and outputs a fixed 512-dimensional signer representation. Importantly, both the MLP and the extracted signer features are kept frozen during test-time adaptation, ensuring that signer information serves as a stable routing prior rather than a learnable shortcut.

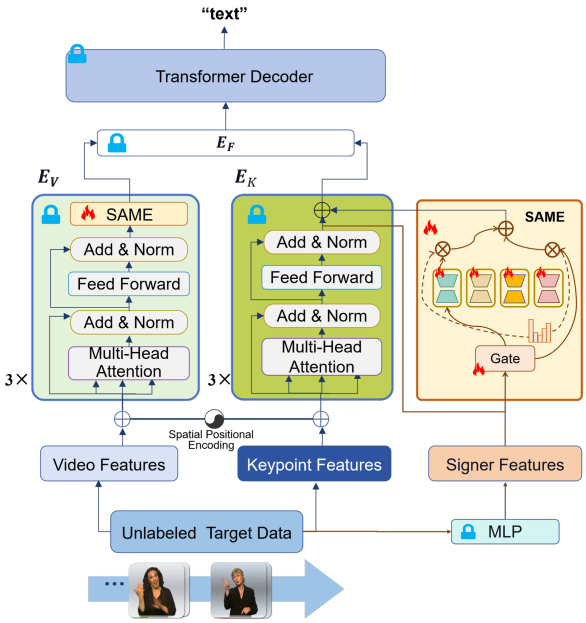


Figure 5: Architecture of the Transformer-based SLT baseline. The model includes video encoder, keypoint encoder, fusion encoder and text decoder. A MLP is used to extract signer features for signer-aware routing. SAME is inserted at the end of each encoder layer. During TTA, only SAME is fine-tuned to adapt to the domain shift.

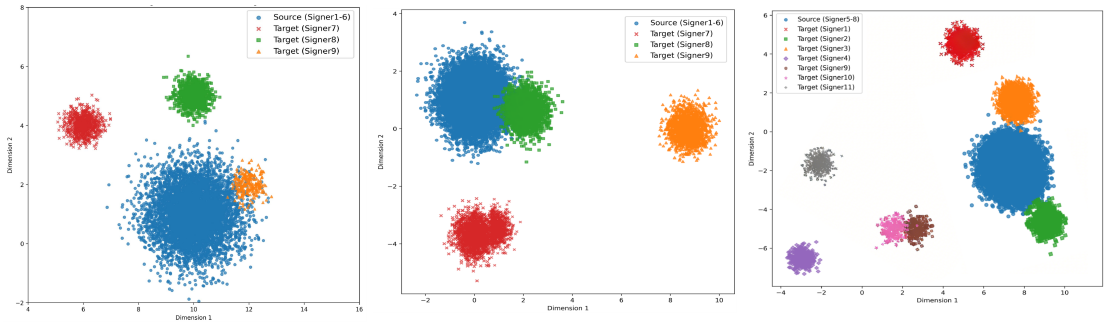
C More Experimental Details

C.1 Datasets

We conduct experiments on three widely used SLT benchmarks, including Phoenix-2014T, CSL-Daily and How2Sign. Phoenix-2014T (Camgoz et al., 2018) consists of German Sign Language broadcast weather recordings aligned with spoken German translations and gloss annotations with 9 signers. CSL-Daily (Zhou et al., 2021a) is a large-scale dataset covering daily conversational topics in Chinese Sign Language and has 10 signers. How2Sign (Duarte et al., 2021) is a multimodal dataset of continuous American Sign Language videos, covering a vocabulary of over 16,000 English words and collected from 11 signers.

C.2 Source–Target Domain Split

We divided each dataset into source domain and target domain according to the signers. For Phoenix-2014T, signers numbered 1–6 are treated as the source domain for model pre-training, and signers 7–9 are regarded as the target domain to evaluate our TTA method. For CSL-Daily, signers 0–6 are set as the source domain and signers 7–9 as the target domain. For How2Sign, since the majority



(a) Phoenix-2014T (b) CSL-Daily (c) How2Sign

Figure 6: Visualization of signer-induced domain shift across datasets

of its data samples are concentrated on signers 5 and 8, we assign signers 5–8 as the source domain, and signers 1–4 and 9–11 as the target domain.

The Phoenix-2014T pre-training dataset has a total of 6120 samples, with the training, validation and test subsets divided into 5211, 384 and 525 samples respectively, and the number of samples used for TTA is 2091. For CSL-Daily, the pre-training dataset includes 15388 samples in total, with the training, validation and test subsets being 13677, 830 and 881 samples respectively, and 5266 samples are used for TTA. The How2Sign pre-training dataset has a total of 28655 samples, where the training, validation and test subsets are 26698, 887 and 1070 samples respectively, and the sample size adopted for TTA is 6474.

The three datasets cover different sign languages and topics, which provide diverse evaluation settings and enable a comprehensive verification of the generality of our method.

C.3 Signer-induced domain shift

We visualize signer features extracted by the source-trained SLT encoder across all datasets. As shown in Fig. 6, target-domain signers consistently exhibit distribution shifts relative to source-domain signers, which indicates that signer variations induce non-negligible domain shift that are not fully resolved by source-domain pre-training, underscoring the necessity of TTA for robust generalization to unseen signers.

C.4 Implementation Detail

The video and keypoint encoder of the SLT baseline used as the source model for TTA consists of three Transformer layers, each equipped with 512-dimensional hidden units and 8 attention heads for multi-head attention. The input representations include RGB and keypoint features with the dimension of 832 per frame. Its decoder follows the same configuration, using a three-layer Transformer decoder with cross-attention to the fused encoder outputs. During test-time adaptation, the parameters of the source SLT model are frozen, and only the SAME modules are updated. The comparison experiments are conducted on both Phoenix-2014T and CSL-Daily datasets, while all ablation and analysis experiments are conducted on Phoenix-2014T.

D More Ablation Studies

In this section, we present more ablation studies on the Phoenix-2014T dataset.

D.1 Detailed Analysis of Adaptation Targets

Fig. 7 illustrates the BLEU-4 scores across adaptation steps for different adaptation targets. Adapting only normalization layers results in marginal improvement, suggesting limited adaptation capacity. Full-parameter tuning achieves stronger performance but requires up to 15 adaptation steps to stabilize, indicating higher optimization difficulty at test time. In contrast, lightweight modules such as LoRA and MoE converge rapidly within 5 steps, exhibiting both higher efficiency and stability. Combining MoE and LoRA (our approach) further improves adaptation effectiveness, achieving the highest BLEU-4 gain with minimal trainable parameters.

D.2 Effect of Signer Features

We evaluate the effect and robustness of signer features by removing signer features, injecting Gaussian noise of varying magnitudes, and replacing signer features with random vectors. As shown in Tab. 9, while noise slightly affects performance, the model still outperforms most baselines and the

Setting	BLEU-1	BLEU-2	BLEU-3	BLEU-4
SAME	**37.61**	**23.94**	**16.43**	**12.77**
w/o signer feature	35.31	21.85	15.17	11.91
Noisy feature (+10%)	36.99	23.08	16.32	12.61
Noisy feature (+20%)	36.76	22.96	16.17	12.49
Noisy feature (+30%)	36.44	22.32	15.98	12.09
Random feature	34.71	21.44	15.08	11.38

Table 9: Impact and robustness analysis of signer features on Phoenix-2014T.

Expert number	Top-k	BLEU-1	BLEU-2	BLEU-3	BLEU-4
6	4	31.47	18.79	12.87	10.21
6	2	32.54	19.55	13.89	11.32
6	1	34.94	20.19	14.71	11.53
5	4	31.55	19.15	13.39	10.58
5	2	33.15	20.98	15.12	11.81
5	1	36.53	21.84	16.28	12.31
4	4	31.67	19.78	13.15	10.64
4	2	36.56	22.72	15.89	12.06
4	1	**37.61**	**23.94**	**16.43**	**12.77**

Table 10: Effect of expert numbers and Top-k on BLEU. (Bold: best results)

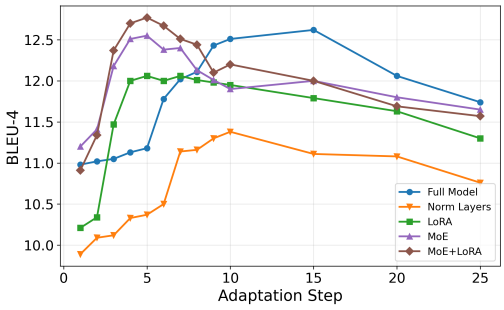


Figure 7: BLEU-4 performance across adaptation steps for different adaptation targets.

version without signer features. In contrast, using random features significantly degrades translation performance, highlighting the effectiveness of signer features.

D.3 Sensitivity Analysis of TTA Loss Weights

To analyze the sensitivity of our method to loss weight choices, we conduct a robustness study by varying the weights of EM, MCC, and PLS. The results are shown in Tab. 11 and Fig. 9. We find that performance remains stable within a practical and reasonable weight range. As shown in Tab. 11, when EM $\in [0.2, 0.4]$, MCC $\in [0.6, 0.8]$, and PLS $\in [0.1, 0.3]$, the BLEU-4 score varies only marginally, indicating that our method is not sensitive to precise hyperparameter tuning in realistic deployment scenarios. When exploring a broad range of weight combinations (Fig. 9), we observe that extreme configurations, such as overly large EM weights or excessively small MCC weights, can degrade performance. This behavior is consistent with prior findings that excessive EM may lead to over-confident and less discriminative predictions.

λ_{EM}	λ_{MCC}	λ_{PLS}	BLEU-4
0.3	0.7	0.2	**12.77**
0.4	0.6	0.2	12.44
0.2	0.8	0.2	12.29
0.3	0.7	0.1	12.63
0.3	0.7	0.3	12.32

Table 11: Sensitivity analysis of TTA loss weights. (Bold: best results)

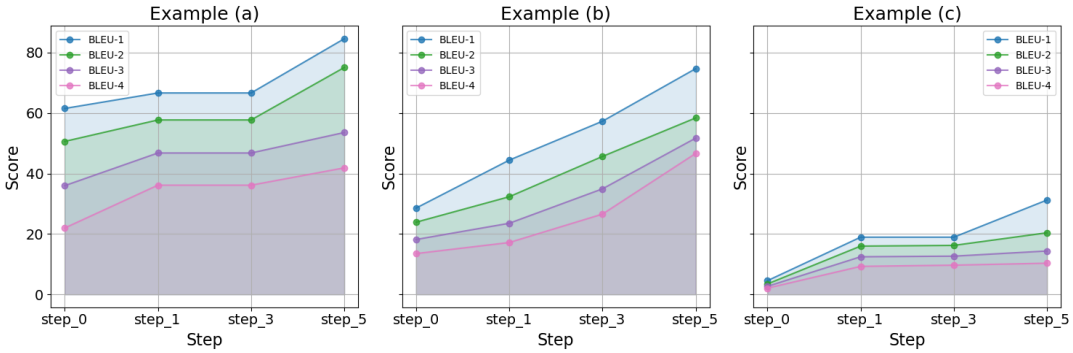


Figure 8: BLEU scores on examples over adaptation step. (Step_0 refers to the model before adaptation)

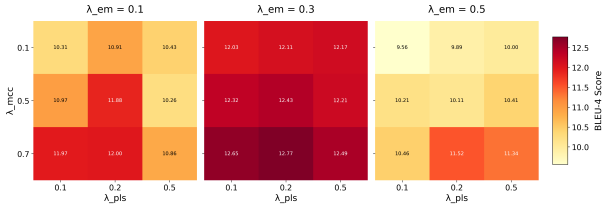


Figure 9: Performance under a wide range of loss weight combinations

D.4 Analysis of the Hyperparameter of Experts

We further investigate the impact of the hyperparameters of the SAME module, including the number of experts and the top-k selection, on the overall TTA performance. As shown in Tab. 10, the best adaptation performance is achieved when using 4 experts with top-1 selection. Increasing the number of activated experts generally leads to performance degradation, likely due to excessive mixing across experts, which weakens domain discrimination and dilutes expert specialization.

E Qualitative Results

This section provides qualitative case studies and visualizes three cases on Phoenix-2014T dataset to demonstrate the effectiveness of our test-time adaptation framework. Fig. 8 and Tab. 12 present the BLEU scores across adaptation steps and the corresponding translation results. The results show that as the adaptation progresses, the BLEU scores continue to increase, and the matching degree between the predicted translation and the actual reference becomes higher and higher, indicating that SAME helps to gradually improve the translation quality under test-time adaptation.

	Example (a)
Ref.	**heute nacht null grad an der ostsee und minus acht grad im bergland.** (Tonight zero degrees on the Baltic Sea and minus eight degrees in the mountains.)
No Adapt	**heute nacht acht grad am alpenrand bis minus acht grad an der küste.** (Tonight eight degrees at the Alpine edge to minus eight degrees at the coast.)
Step-1	**heute nacht acht grad an der ostsee und bis minus acht grad an den alpen.** (Tonight eight degrees on the Baltic Sea and up to minus eight degrees in the Alps.)
Step-3	**heute nacht acht grad an der ostsee und null grad im bayerischen wald.** (Tonight eight degrees on the Baltic Sea and zero degrees in the Bavarian Forest.)
Step-5	**heute nacht null grad an der ostsee und minus acht grad im bayerischen wald.** (Tonight zero degrees on the Baltic Sea and minus eight degrees in the Bavarian Forest.)
	Example (b)
Ref.	**heute nacht ist es im süden teils klar teilweise aber auch nebel.** (Tonight it will be partly clear in the south, but there will also be some fog.)
No Adapt	**heute nacht ist es klar in der südhälfte teils klar teils klar teils klar teils klar teils klar teils locker bewölkt.** (The weather is sunny today, partly sunny, partly sunny, partly sunny, partly cloudy.)
Step-1	**heute nacht ist es nördlich des mains verbreitet klar oder nur locker bewölkt im süden bildet sich nebel.** (The weather will be sunny or cloudy tonight, with fog in the south.)
Step-3	**heute nacht ist es meist bewölkt im süden stellenweise nebel** (Tonight it will be mostly cloudy in the south with some fog in places.)
Step-5	**heute nacht ist es im süden teilweise klar und gebietsweise nebel.** (Tonight it will be partly clear in the south, with areas of fog.)
	Example (c)
Ref.	**am tag weht der wind im nordwesten frisch dort kann er böen und sturmstärke erreichen.** (During the day the wind in the northwest blows fresh and may reach gusts and storm strength.)
No Adapt	**vor allem in der westhälfte mitunter sturmböen.** (Especially in the west, there are sometimes storms.)
Step-1	**vor allem in der norden weht der wind schwach bis mäßig.** (Especially in the north, the wind blows weak to moderate.)
Step-3	**vor allem in der nordwesthälfte weht der wind schwach bis mäßig.** (Especially in the northwestern, the wind blows weak to moderate.)
Step-5	**vor allem in der nordwesthälfte weht der wind mäßig bis frisch mit starken bis stürmischen böen.** (Especially in the northwestern, the wind blows moderate to fresh with strong to stormy gusts.)

Table 12: Qualitative results on Phoenix-2014T