TOWARDS MULTI-VIEW SIGN LANGUAGE UNDERSTANDING: A BENCHMARK DATASET AND BASELINE

Anonymous authors
Paper under double-blind review

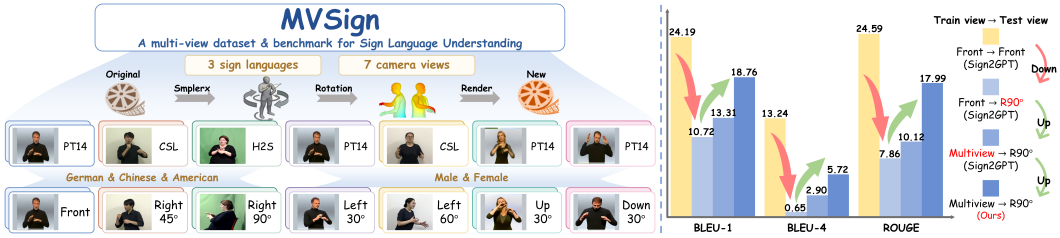


Figure 1: Overview of MVSign benchmark and viewpoint robustness analysis. **Left:** MVSign combines original recordings with synthesized views to cover three sign languages and seven viewpoints. **Right:** A controlled comparison on How2Sign with a fixed training-set size. Sign2GPT degrades when testing shifts from frontal videos to paired native $R90°$ recordings. Replacing 40% of frontal training samples with multi-view samples improves side-view performance, while CanonSLU achieves further gains using the same multi-view training data and $R90°$ test set.

ABSTRACT

Most existing Sign Language Understanding (SLU) methods are developed under fixed-view settings and remain vulnerable to viewpoint changes, while the scarcity of multi-view data hinders systematic research on viewpoint robustness. To address this gap, we introduce **MVSign**, a benchmark spanning three sign languages and seven viewpoints. MVSign combines original frontal recordings from existing datasets with additional views synthesized via 3D whole-body motion reconstruction and target-view rendering, while preserving the source splits and annotations. We further propose **CanonSLU**, a framework for multi-view SLU. This framework employs a canonical-view-guided strategy, using the frontal view as a semantic anchor. Specifically, CanonSLU incorporates *Canonical-View Knowledge Transfer (CKT)* to transfer canonical knowledge from the frontal view to non-frontal views, and *Motion Relation Aggregation (MRA)* to aggregate motion-related features across frames and strengthen temporal representations under viewpoint changes. Extensive experiments on MVSign demonstrate competitive performance with gains on non-frontal views, establishing CanonSLU as an effective baseline for multi-view SLU.

1 INTRODUCTION

Sign Language Understanding (SLU) (Li et al., 2025) converts sign language videos into gloss sequences or spoken-language sentences, serving as an important technique for improving communication accessibility between the deaf community and the broader information environment. SLU mainly encompasses Isolated Sign Language Recognition (ISLR) (Pu et al., 2024; Wang et al., 2025a), Continuous Sign Language Recognition (CSLR) (Hu et al., 2023b; Gan et al., 2023; Xu et al., 2025) and Sign Language Translation (SLT) Wong et al. (2024); Gan et al. (2025); Guo et al. (2025). ISLR classifies individual sign movements, whereas CSLR learns the temporal alignment between continuous sign language sequences and their corresponding glosses. SLT further translates sign language videos into natural language sentences. Despite substantial progress on standard CSLR and SLT benchmarks (Huang et al., 2018; Camgoz et al., 2018; Zhou et al., 2021), driven by advances in visual representation learning (Koller et al., 2019; Min et al., 2021), sequence modeling (Hao et al., 2021; Gan et al., 2024; Cheng et al., 2020), and large-scale pre-training (Chen et al., 2022a; Hu et al., 2023a; Li et al., 2025), most existing systems are developed and evaluated under nearly fixed-view settings, where training and testing videos are captured from the same or very

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

similar camera positions. Consequently, their performance on current benchmarks may overestimate real-world robustness, leaving ***viewpoint variation*** as a practical yet underexplored challenge.

We conduct a controlled experiment on How2Sign (Duarte et al., 2021) with the representative SLT model Sign2GPT (Wong et al., 2024), using paired frontal and native $R90^\circ$ recordings (Figure 1 (right)). With the training-set size fixed, replacing the frontal-view test samples with their $R90^\circ$ side-view counterparts leads to a large performance drop, because viewpoint changes alter hand appearance, motion projection, occlusion, and hand-body coordination. Replacing 40% of the frontal training samples with multi-view samples improves side-view performance under the same training budget, showing that broader viewpoint coverage provides useful training signals for cross-view SLU. Moreover, CanonSLU achieves further gains on the same multi-view training data and $R90^\circ$ test set, suggesting that robust multi-view SLU requires not only viewpoint coverage but also models that exploit cross-view semantic and motion cues. However, such coverage is scarce in continuous SLU: mainstream benchmarks such as PHOENIX-2014T (Camgoz et al., 2018) and CSL-Daily (Zhou et al., 2021) mainly provide frontal-view videos; existing large-scale multi-view datasets (Li et al., 2020; Desai et al., 2023; Shen et al., 2024; Dinh et al., 2025) primarily target ISLR; and How2Sign only provides frontal and $R90^\circ$ views without gloss annotations. Collecting synchronized multi-camera data with sequence-level annotations is also costly. These observations point to the need for multi-view data that can be obtained reliably and efficiently from existing single-view recordings, as well as models tailored to multi-view SLU.

To address this need, we introduce **MVSign**, a multi-view SLU benchmark spanning three sign languages and seven viewpoints. As illustrated in Figure 1 (left), MVSign comprises MVSign-PT14, MVSign-CSL, and MVSign-H2S. It uses the frontal view as a canonical reference shared by all sequences and extends it with six non-frontal views $\{R90^\circ, R45^\circ, L30^\circ, L60^\circ, U30^\circ, D30^\circ\}$. For sources without native multi-view recordings, new views are constructed around the signer, not the camera: each recording is first lifted into a unified 3D whole-body motion representation, which then guides the re-rendering of new viewpoints into realistic videos, preserving the source splits, annotations, and motion content. To improve viewpoint robustness in multi-view SLU, we further propose **CanonSLU**, a canonical-view-guided framework that treats the frontal view as a semantic anchor. CanonSLU incorporates *Canonical-View Knowledge Transfer (CKT)* to transfer canonical knowledge from a pretrained frontal-view model to non-frontal inputs, and *Motion Relation Aggregation (MRA)* to aggregate motion-aware relations across frames and strengthen temporal representations under viewpoint changes. Our contributions are summarized as follows:

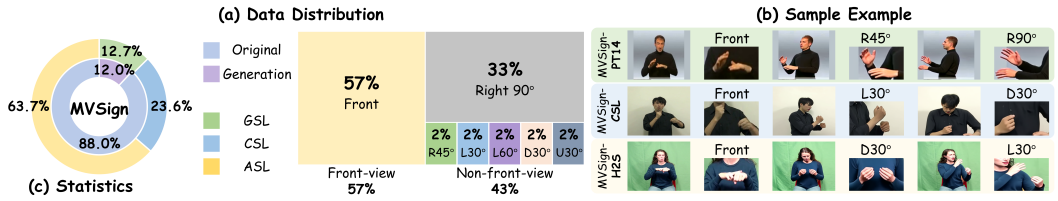
(1) We construct **MVSign**, a multi-view SLU benchmark covering three sign languages and seven viewpoints, combining synthesized and naturally captured multi-view videos while preserving original splits, annotations, and motion sequences.

(2) We propose **CanonSLU**, a canonical-view-guided framework that uses the frontal view as a semantic anchor, integrating CKT for frontal-to-non-frontal knowledge transfer and MRA for motion-aware temporal aggregation under viewpoint changes.

(3) Extensive experiments on MVSign across CSLR and SLT show that CanonSLU improves performance under viewpoint variations, with gains on non-frontal views, establishing a strong baseline for multi-view SLU.

2 MVSIGN DATASET

To expand viewpoint coverage for continuous SLU, we construct **MVSign**, a multi-view benchmark comprising MVSign-PT14, MVSign-CSL, and MVSign-H2S, derived from PHOENIX-2014T, CSL-Daily, and How2Sign, respectively. We retain all original frontal-view videos as the *Front* view and preserve the native *R90°* recordings in MVSign-H2S. Using the three-stage construction process in Figure 3, we generate 1,000 videos for each of six non-frontal viewpoints (*R45°*, *R90°*, *L30°*, *L60°*, *U30°*, and *D30°*) for both MVSign-PT14 and MVSign-CSL, yielding 6,000 additional videos per subset. For each of the five additional viewpoints in MVSign-H2S, excluding *R90°*, we sample 100 source videos from each original train, dev, and test split and apply the same construction process, yielding 300 generated videos per viewpoint. All generated videos inherit their source split assignments and linguistic annotations: gloss and sentence annotations for MVSign-PT14 and MVSign-CSL, and sentence annotations for MVSign-H2S. Figure 2 summarizes the data distribution, examples, and statistics of MVSign.



Datasets	Language	Signers	Annotation	Angle types	Train	Dev	Test	Total
MVSign-PT14	German Sign Language (GSL)	9	text, gloss	*front, R90°, L60°, R45°, L30°, D30°, U30°	11896	1119	1242	14257
MVSign-CSL	Chinese Sign Language (CSL)	10	text, gloss	*front, R90°, L60°, R45°, L30°, D30°, U30°	23201	1677	1776	26654
MVSign-H2S	American Sign Language (ASL)	11	text	*front, *R90°, L60°, R45°, L30°, D30°, U30°	62594	3974	5186	71754

Figure 2: Overview of the MVSign dataset. (a) Data distribution over original vs. generated, sign languages and viewpoints. (b) Sample multi-view examples. (c) Statistics of dataset.

Whole-body Parameter Extraction. Given an input single-view RGB video, we first recover a 3D parameterized whole-body representation for each frame using SMPLer-X (Cai et al., 2023), which is pretrained on multiple human behavior datasets (*e.g.*, 3DPW (Von Marcard et al., 2018), COCO (Jin et al., 2020), InterHand2.6M (Moon et al., 2020) and UBody (Lin et al., 2023)). The estimated parameters are denoted by $\theta_s = \{\theta_b, \theta_h, \theta_e, \theta_j, \theta_{\text{root}}, \theta_{\text{shape}}, \psi\}$, where θ_b, θ_h, θ_e, θ_j, θ_{root}, θ_{shape}, ψ represent the body pose, bilateral hand pose, eye pose, jaw pose, global root orientation, body shape, and facial expression parameters, respectively.

Based on these parameters, we reconstruct the intermediate mesh sequence. Compared with direct pixel-space manipulation, this parameterized 3D representation provides a more geometry-consistent basis for view synthesis and better preserves fine-grained sign-specific motion patterns under fast articulation, self-occlusion, and hand–body interaction.

Root-Driven Viewpoint Transformation. After recovering the 3D body representation, we synthesize new viewpoints by transforming the body in 3D space rather than manipulating images directly. We introduce viewpoint changes by modifying the global root (θ_{root}) orientation while retaining the estimated local pose, body-shape, and facial-expression parameters for each frame. Let $\mathbf{V} \in \mathbb{R}^{3\times M}$ denote the mesh vertices of a frame, where M is the number of mesh vertices. Given a target viewpoint parameterized by yaw angle α and pitch angle β, the transformed mesh is defined as:

$$\mathbf{V}' = \mathbf{R}_{\text{view}}\mathbf{V} = \mathbf{R}_y(\alpha)\mathbf{R}_x(\beta)\mathbf{V} = \begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\beta & -\sin\beta \\ 0 & \sin\beta & \cos\beta \end{bmatrix} \mathbf{V}. \tag{1}$$

The same target-view rotation is applied across all frames, preserving the local joint configurations and relative temporal dynamics among the torso, arms, and hands of the recovered motion.

Skeleton-Guided Adaptive Rendering. To obtain realistic sign language videos after viewpoint transformation, we first render the viewpoint-transformed mesh sequence to obtain an intermediate video. DWPose (Yang et al., 2023) extracts a 2D skeleton sequence from the rendered frames, as motion guidance for StableAnimator (Tu et al., 2025). Conditioned on this skeleton sequence and identity cues from the original video, StableAnimator generates the target-view sign language video.

3 Method

3.1 Preliminaries

Given a sign language video captured under viewpoint v, we denote the input sequence as $X_i^v = \{x_{i,t}^v\}_{t=1}^{T_i} \in \mathbb{R}^{T_i\times 3\times H\times W}$, where i indexes the sign sample, T_i is the number of frames, and H and W denote the frame height and width, respectively. $v \in \mathcal{V}$ belongs to a predefined set of viewpoints. We use $f \in \mathcal{V}$ to denote the frontal view, which serves as the canonical view in our framework. Depending on the available annotations, each sample may be associated with a gloss sequence $G_i = \{g_{i,m}\}_{m=1}^{M_i}$ and a spoken-language sentence $S_i = \{s_{i,n}\}_{n=1}^{N_i}$. In this work, we study multi-view SLU under three task settings: CSLR, SLT (including Gloss-based SLT and Gloss-free SLT). (1) **CSLR** maps a sign video X_i^v to its corresponding gloss sequence G_i. (2) **Gloss-based SLT** translates X_i^v into a spoken-language sentence S_i using gloss annotations as supervision during training. (3) **Gloss-free SLT** translates X_i^v into S_i without using manually annotated glosses.

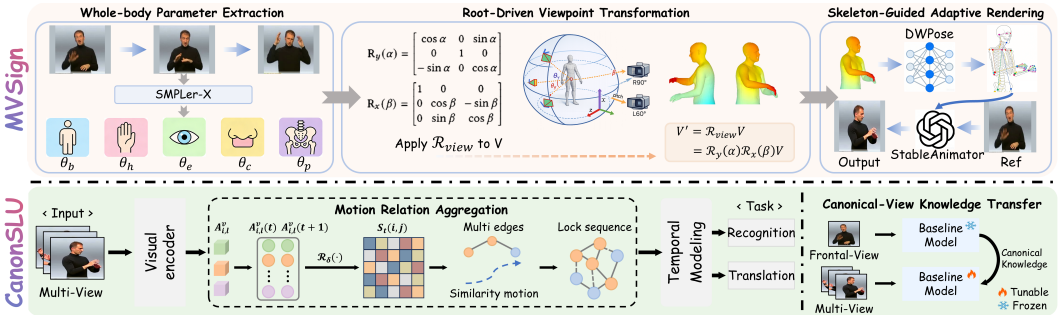


Figure 3: Overview of **MVSign** and **CanonSLU** framework. The upper row illustrates the three-stage construction process used to extend multi-view dataset. The lower row presents CanonSLU, which employs a shared visual-temporal encoder, **Motion Relation Aggregation**, and **Canonical-View Knowledge Transfer** with task-specific CSLR and SLT heads.

3.2 CanonSLU Framework

Since viewpoint changes perturb hand appearance, motion projection, and occlusion patterns, they cause both representation-level disturbance and prediction-level semantic discrepancy. As shown in Figure 3, we build a framework for multi-view SLU, where a shared visual-temporal encoder supports both gloss and sentence prediction. CanonSLU consists of three components: (i) **Baseline Model** that extracts visual-temporal features and performs CSLR or SLT with task-specific prediction heads, (ii) Motion Relation Aggregation (**MRA**) for motion-aware representation learning within each view, (iii) Canonical-View Knowledge Transfer (**CKT**) that uses the frontal view as a semantic anchor and transfers canonical knowledge to non-frontal views when distillation is used.

Baseline Model. Given a sign video X_i^v captured under viewpoint v, we first extract frame-wise visual features with a visual encoder:

$$F_i^v = E_{\text{vis}}(X_i^v). \tag{2}$$

The visual features are then processed by a temporal convolution module and a BiLSTM, which capture short-range temporal dynamics and long-range temporal dependencies, respectively:

$$\widetilde{H}_i^v = \text{CNN}_{\text{1D}}(F_i^v), \qquad H_i^v = \text{BiLSTM}(\widetilde{H}_i^v). \tag{3}$$

For **CSLR**, both the BiLSTM output H_i^v and the convolution-level feature $\widetilde{H}_i^v$ are projected into the gloss vocabulary by the gloss classifier:

$$\widetilde{Z}_i^v = C_{\text{gloss}}(\widetilde{H}_i^v), \qquad Z_i^v = C_{\text{gloss}}(H_i^v). \tag{4}$$

Here, Z_i^v denotes the final sequence logits for gloss prediction, while $\widetilde{Z}_i^v$ provides auxiliary logits for intermediate supervision. The final gloss sequence is directly recognized from Z_i^v:

$$\hat{G}_i = \text{Recognition}(Z_i^v). \tag{5}$$

For **SLT**, the BiLSTM feature H_i^v is first mapped into the mBART embedding space by a visual-language mapper, implemented as a two-layer MLP, which builds a connection between the two different modalities. The mapped feature is then fed into mBART to obtain translation logits:

$$T_i^v = \text{mBART}\left(\text{Mapper}(H_i^v)\right). \tag{6}$$

Here, T_i^v denotes the token-level logits over the text vocabulary. The final spoken sentence is directly translated from T_i^v:

$$\hat{S}_i = \text{Translation}(T_i^v). \tag{7}$$

Motion Relation Aggregation. To improve viewpoint robustness at the representation level, we augment the visual encoder with **MRA**. The motivation is that viewpoint changes mainly affect local appearance and spatial projection, while the underlying action sequence remains unchanged. Therefore, we model motion relations across frames and aggregate relevant visual features, aiming to capture more stable action cues across viewpoints for SLU. Let $A_{i,l}^v \in \mathbb{R}^{C_l \times T_i \times H_l \times W_l}$ denote the

intermediate feature map from the l-th stage of the visual encoder E_{vis}. We first compute frame-wise motion features by temporal differencing:

$$M^v_{i,l}(t) = A^v_{i,l}(t) - A^v_{i,l}(t-1). \tag{8}$$

Using the refined motion features and projected value features, MRA aggregates motion-related information across multiple temporal offsets:

$$O^v_{i,l} = \text{Agg}_{\delta\in\mathcal{T}_{\text{off}}}\, \mathcal{R}_\delta\left(\phi_m(M^v_{i,l}), \phi_v(A^v_{i,l})\right), \qquad \mathcal{T}_{\text{off}} = \{1, 2, 4\}. \tag{9}$$

where $\phi_m(\cdot)$ is a temporal convolution for motion refinement, $\phi_v(\cdot)$ is a $1 \times 1 \times 1$ value projection, and $\mathcal{R}_\delta(\cdot)$ denotes motion relational aggregation over the temporal offsets δ. Specifically, it computes motion similarity between frames separated by each δ, selects the most related spatial positions, and aggregates their value features in both forward and backward temporal directions.

Finally, the aggregated feature $O^v_{i,l}$ is selectively injected into the original feature map through a motion-aware gated residual update:

$$\widehat{A}^v_{i,l} = A^v_{i,l} + \alpha_l \left[\phi_o(O^v_{i,l}) \odot G^v_{i,l}\right]. \tag{10}$$

Here, ϕ_o is an output temporal convolution, and $G^v_{i,l}$ is a sigmoid gate computed from the concatenation of $A^v_{i,l}$ and $\phi_m(M^v_{i,l})$. The operator $\odot$ denotes element-wise multiplication, and α_l is a learnable residual weight. In our implementation, MRA is applied at stages $l \in \{3, 4\}$ of the visual encoder, where the lower spatial resolution reduces the cost of dense cross-frame matching. The aggregation and gating operations are detailed in Appendix B.

Canonical-View Knowledge Transfer. Although MRA enhances motion-aware representations within each view, different viewpoints may still produce semantic biases. To reduce this discrepancy, we use the frontal view f as the canonical semantic anchor and transfer its knowledge to non-frontal views. Given paired frontal and non-frontal videos, we compute distillation at corresponding prediction positions and average the loss over valid positions:

$$\mathcal{D}_{\text{KD}}(\mathcal{Q}^f_i, \mathcal{Q}^v_i) = \mathbb{I}[v \neq f] \frac{T_d^2}{\sum_{r=1}^{L_i} m_{i,r}} \sum_{r=1}^{L_i} m_{i,r} \cdot D_{\text{KL}}(\sigma(\mathcal{Q}^f_{i,r}/T_d) \parallel \sigma(\mathcal{Q}^v_{i,r}/T_d)), \tag{11}$$

where $\mathcal{D}_{\text{KD}}(\cdot,\cdot)$ denotes the logit-level distillation loss averaged over valid sequence positions. $\mathcal{Q}^f_{i,r}$ and $\mathcal{Q}^v_{i,r}$ are the teacher and student logits at position r, respectively. L_i denotes the sequence length including padding, and $m_{i,r}$ indicates a valid position. The softmax $\sigma(\cdot)$ is applied over the task-specific vocabulary, T_d is the distillation temperature. The position index r refers to encoder time steps for CSLR and target-token positions for gloss-free SLT. Teacher logits are detached from the computation graph, and $\mathbb{I}[v \neq f]$ activates distillation for non-frontal samples.

3.3 Training Objective

CSLR. We first train a frontal-view recognition model using only frontal-view samples. The objective contains CTC losses on both the final sequence logits and the auxiliary convolutional logits:

$$\mathcal{L}^{\text{front}}_{\text{CSLR}} = \mathcal{L}_{\text{CTC}}(Z^f_i, G_i) + \mathcal{L}_{\text{CTC}}(\widetilde{Z}^f_i, G_i). \tag{12}$$

Then, we train the model with all views and transfer frontal-view knowledge to non-frontal views:

$$\mathcal{L}^{\text{multi}}_{\text{CSLR}} = \mathcal{L}_{\text{CTC}}(Z^v_i, G_i) + \mathcal{L}_{\text{CTC}}(\widetilde{Z}^v_i, G_i) + \lambda \mathcal{D}_{\text{KD}}(Z^f_i, Z^v_i), \tag{13}$$

where Z^f_i denotes the pretrained sequence logits from the frontal-view recognition model.

Gloss-based SLT. We initialize the visual-temporal encoder from the multi-view CSLR model trained with CKT and the translation module from mBART-large-cc25. Both modules are then jointly fine-tuned on all views using sentence-level cross-entropy and two gloss-level CTC losses:

$$\mathcal{L}^{\text{multi}}_{\text{GBSLT}} = \mathcal{L}_{\text{CE}}(T^v_i, S_i) + \mathcal{L}_{\text{CTC}}(Z^v_i, G_i) + \mathcal{L}_{\text{CTC}}(\widetilde{Z}^v_i, G_i). \tag{14}$$

Gloss-free SLT. Following MixSignGraph Gan et al. (2025), we train the frontal-view translation model with pseudo-gloss supervision. The pseudo glosses $\widetilde{G}_i$ are derived from the target sentence S_i by punctuation removal and lemmatization, requiring no manual gloss annotations:

$$\mathcal{L}^{\text{front}}_{\text{GFSLT}} = \mathcal{L}_{\text{CE}}(T^f_i, S_i) + \mathcal{L}_{\text{CTC}}(Z^f_i, \widetilde{G}_i) + \mathcal{L}_{\text{CTC}}(\widetilde{Z}^f_i, \widetilde{G}_i). \tag{15}$$

During multi-view training, we supervise translation with the reference sentences using cross-entropy, and distill the translation logits from the frontal teacher at the corresponding target-token positions under teacher forcing, where both decoders receive the same ground-truth target prefix:

$$\mathcal{L}_{\text{GFSLT}}^{\text{multi}} = \mathcal{L}_{\text{CE}}(T_i^v, S_i) + \lambda \mathcal{D}_{\text{KD}}(T_i^f, T_i^v), \tag{16}$$

where T_i^f denotes the pretrained translation logits from the frontal-view translation model. Pseudo-gloss CTC provides auxiliary alignment during frontal-view pretraining; since pseudo glosses are automatically derived rather than manually annotated, multi-view training uses $\mathcal{L}_{\text{CE}}$ and $\mathcal{D}_{\text{KD}}$.

4 Experiments

Experimental Settings. All experiments are conducted on MVSign. MVSign-H2S is evaluated only for gloss-free SLT due to the absence of gloss annotations. We use WER for CSLR task and BLEU-1–4 and ROUGE for SLT task. Detailed experimental settings are provided in Appendix C.

4.1 Discussion on Data Quality

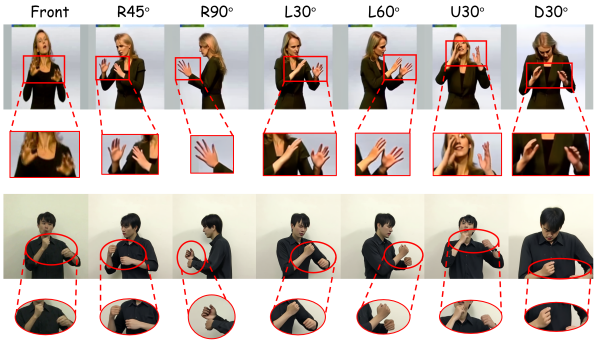


Figure 4: Visualization of the same sign sample under multiple viewpoints with evident appearance variations.

Qualitative Assessment of Visual Fidelity. Figure 4 shows two sign sequences from seven viewpoints. The red annotations and enlarged crops highlight hand region, where viewpoint changes have a strong influence on visual appearance. Across the views, the generated videos preserve coherent body contours and signer appearance while maintaining fine-grained hand details. In particular, individual fingers remain visually distinguishable in non-frontal viewpoints and in configurations involving partial self-occlusion or closely grouped fingers, without obvious finger merging or structural artifacts. These examples provide evidence that the rendering process preserves local visual cues relevant to sign recognition under viewpoint changes.

Table 1: Pose fidelity in MVSign.

	NME↓	PCK@0.05↑	PCK@0.10↑	TJ-Err↓
Body	0.0132	0.9814	0.9962	0.0075
Left Hand	0.0233	0.9438	0.9835	0.0210
Right Hand	0.0139	0.9844	0.9971	0.0164

Quantitative Pose-Fidelity Evaluation. The synthesized videos are generated by rotating the recovered 3D body around the global root, thereby preserving local joint configurations and temporal motion. We assess geometric fidelity by projecting the transformed 3D joints under each target-view and comparing these reference poses with 2D body and hand keypoints re-extracted from the rendered videos using DWPose. Because the references are derived directly from the transformed 3D motion, this evaluation measures consistency between the target geometry and the rendered observations. Averaged over the non-frontal viewpoints, the results in Table 1 yield NME $\leq$ 0.0233, PCK@0.05 $\geq$ 0.9438, PCK@0.10 $\geq$ 0.9835, and TJ-Err (Temporal Jitter Error) $\leq$ 0.0210, supporting the fidelity of motion for multi-view SLU.

4.2 Comparison with State-of-the-Art Methods on CSLR Task

Table 2 compares representative state-of-the-art CSLR methods on the proposed MVSign-PT14 and MVSign-CSL subsets. For fair comparison, all competing methods are retrained on these two multi-view datasets following the training protocols and implementation details described in their original papers. **CanonSLU** achieves the lowest WER on all evaluation splits.

MVSign-PT14. CanonSLU obtains WERs of 33.27% and 33.35% on the dev and test sets, respectively. Compared with the strongest baseline, SignGraph (35.13%) and CorrNet (34.80%) on the test set, CanonSLU reduces WER by **1.78** and **1.45**, respectively. These results demonstrate the consistent effectiveness of CanonSLU for CSLR under multi-view settings.

Table 2: Comparison of CSLR performance on the MVSign dataset.

Methods	MVSign-PT14				MVSign-CSL			
	Dev-WER↓	Dev-del↓/ins↓	Test-WER↓	Test-del↓/ins↓	Dev-WER↓	Dev-del↓/ins↓	Test-WER↓	Test-del↓/ins↓
SMKD (Hao et al., 2021)	42.83	17.61/2.06	42.91	16.06/2.68	45.78	16.56/2.23	45.01	16.24/2.24
VAC (Min et al., 2021)	39.85	16.20/2.35	39.57	16.21/3.21	46.62	18.31/2.41	44.59	17.15/2.46
MMTLB (Chen et al., 2022a)	38.89	12.88/4.33	38.05	11.55/4.57	44.28	16.44/3.71	43.28	15.02/3.74
C^2SLR (Zuo & Mak, 2022)	41.90	19.10/2.40	40.50	18.50/2.9	46.64	18.31/1.96	45.22	18.08/1.89
SEN (Hu et al., 2023c)	37.77	14.77/2.61	38.17	14.18/3.34	42.23	13.08/3.36	41.11	12.25/3.31
CorrNet (Hu et al., 2023b)	34.93	12.50/2.36	34.80	11.61/2.95	41.21	13.47/3.41	40.21	12.54/2.95
SignGraph (Gan et al., 2024)	34.65	11.58/2.65	35.13	10.52/3.53	38.63	13.50/2.93	37.63	12.79/2.59
HSTE (Xu et al., 2025)	42.03	17.21/2.11	40.31	18.23/2.95	44.33	16.32/3.73	42.74	14.23/3.55
MSKA (Guan et al., 2025)	35.25	11.96/2.51	35.45	10.78/3.10	38.07	14.08/2.73	37.39	12.99/2.47
CanonSLU (Ours)	**33.27**	12.95/2.25	**33.35**	11.85/2.89	**36.92**	14.53/2.39	**35.71**	13.68/2.27

Table 3: Comparison of SLT performance on the MVSign dataset.

Task	Methods	DEV					TEST				
		BLEU1↑	BLEU2↑	BLEU3↑	BLEU4↑	ROUGE↑	BLEU1↑	BLEU2↑	BLEU3↑	BLEU4↑	ROUGE↑
	MVSign-PT14										
Gloss-based	MMTLB (Chen et al., 2022a)	44.55	31.56	24.20	19.55	44.49	43.34	30.84	23.60	19.04	43.76
Gloss-based	SLTUNET (Zhang et al., 2023)	45.61	33.50	25.83	20.90	46.01	43.96	30.85	23.91	18.97	44.79
Gloss-based	CV-SLT (Zhao et al., 2024)	45.36	32.45	25.79	20.80	46.67	44.06	30.02	23.55	18.68	44.38
Gloss-based	MSKA (Guan et al., 2025)	44.78	31.45	23.85	18.96	43.51	43.98	30.93	23.31	18.35	42.40
Gloss-based	**CanonSLU (Ours)**	**46.66**	**33.68**	**26.11**	**21.19**	**46.89**	**44.54**	**31.98**	**24.53**	**19.73**	**45.47**
Gloss-free	GFSLT-VLP (Zhou et al., 2023)	31.67	22.31	17.09	13.78	31.83	28.90	19.80	14.71	11.75	29.62
Gloss-free	Sign2GPT (Wong et al., 2024)	35.61	24.60	18.34	14.41	35.30	33.94	23.62	17.55	13.84	34.20
Gloss-free	SpaMo (Hwang et al., 2025)	36.30	23.24	17.05	13.52	30.92	35.50	22.70	16.44	12.99	29.18
Gloss-free	MMSLT (Kim et al., 2025)	37.40	26.46	20.13	15.06	36.11	35.95	25.54	19.46	14.68	34.84
Gloss-free	**CanonSLU (Ours)**	**41.18**	**27.84**	**20.32**	**15.66**	**42.27**	**39.48**	**26.85**	**19.78**	**15.33**	**40.77**
	MVSign-CSL										
Gloss-based	MMTLB (Chen et al., 2022a)	42.75	30.11	21.75	16.29	40.65	42.65	30.08	21.79	16.35	40.74
Gloss-based	SLTUNET (Zhang et al., 2023)	40.67	28.32	20.62	15.55	39.63	40.94	28.59	20.80	15.69	40.57
Gloss-based	CV-SLT (Zhao et al., 2024)	41.32	29.44	20.58	16.01	40.73	42.66	29.82	21.27	16.57	41.10
Gloss-based	MSKA (Guan et al., 2025)	42.00	28.48	21.77	16.07	40.26	42.37	28.13	21.47	16.74	40.34
Gloss-based	**CanonSLU (Ours)**	**43.46**	**30.75**	**22.76**	**17.47**	**41.42**	**44.43**	**31.60**	**23.51**	**18.12**	**42.04**
Gloss-free	GFSLT-VLP (Zhou et al., 2023)	17.67	7.72	4.49	3.06	18.15	17.14	7.16	4.00	2.58	17.57
Gloss-free	Sign2GPT (Wong et al., 2024)	35.38	22.96	14.88	9.87	33.84	35.96	23.28	14.99	9.78	34.21
Gloss-free	SpaMo (Hwang et al., 2025)	29.32	18.07	12.23	8.74	27.83	29.60	17.93	12.13	8.72	27.06
Gloss-free	MMSLT (Kim et al., 2025)	36.32	23.13	14.02	10.38	36.79	35.94	23.54	14.65	10.66	36.72
Gloss-free	**CanonSLU (Ours)**	**39.91**	**27.22**	**19.67**	**14.90**	**38.04**	**40.93**	**28.11**	**20.39**	**15.47**	**38.84**
	MVSign-H2S										
Gloss-free	GFSLT-VLP (Zhou et al., 2023)	14.76	9.46	4.83	1.93	13.37	14.23	8.98	4.72	2.06	13.51
Gloss-free	Sign2GPT (Wong et al., 2024)	16.61	11.32	7.04	4.55	16.47	16.70	11.01	7.11	4.63	16.93
Gloss-free	SpaMo (Hwang et al., 2025)	17.21	12.38	7.17	4.24	17.83	17.06	11.58	7.74	4.33	17.99
Gloss-free	**CanonSLU (Ours)**	**20.32**	**14.87**	**9.61**	**6.19**	**20.71**	**20.77**	**14.42**	**8.98**	**6.62**	**20.90**

MVSign-CSL. CanonSLU obtains WERs of 36.92% and 35.71% on the dev and test sets, respectively. MSKA provides the strongest baseline performance on both splits, while CanonSLU reduces WER by **1.15** and **1.68** on the dev and test sets, respectively. These results further confirm the effectiveness of CanonSLU for robust multi-view CSLR.

4.3 Comparison with State-of-the-Art Methods on SLT Task

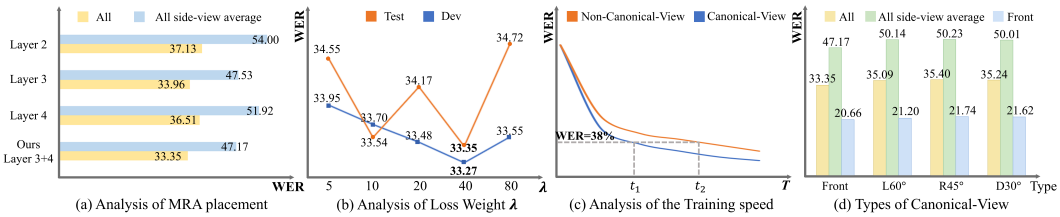
Table 3 compares representative state-of-the-art SLT methods under gloss-based and gloss-free settings on MVSign. We follow the same evaluation protocol as in the CSLR experiments to ensure a fair comparison. **CanonSLU** achieves the best performance across all datasets and evaluation splits.

MVSign-PT14. CanonSLU achieves the best performance under both gloss-based and gloss-free settings. With gloss supervision, it obtains BLEU-4 scores of 21.19 and 19.73 on the dev and test sets, respectively, and ranks first across all metrics. On the dev set, it surpasses MSKA by **2.23** BLEU-4 and **3.38** ROUGE. Without gloss supervision, CanonSLU achieves ROUGE scores of 42.27 and 40.77 on the dev and test sets, respectively, maintaining strong performance.

MVSign-CSL. CanonSLU ranks first across all metrics under both supervision settings. With gloss supervision, it achieves BLEU-4/ROUGE scores of 18.12/42.04 on the test set, exceeding the strongest competing results by **1.38/0.94**, respectively. Without gloss supervision, it obtains BLEU-4/ROUGE scores of 15.47/38.84, improving over MMSLT by **4.81/2.12** on the test set, respectively. CanonSLU thus maintains its advantage under both supervision settings.

Table 4: Ablation results of different module combinations for CSLR on the MVSign-PT14. "All" represents all data; "R", "L", "D", and "U" represent right rotation, left rotation, downward tilt and upward tilt, respectively. Δ indicates the performance change compared with the Baseline.

Methods			Multi-view		Large angle			Small angle			Pitch			Front	
Baseline	CKT	MRA	All	Δ	R90°	L60°	Δ_{Avg}	R45°	L30°	Δ_{Avg}	D30°	U30°	Δ_{Avg}	0°	Δ
✓			37.73	-	58.04	50.73	-	53.91	52.52	-	55.05	52.10	-	22.99	-
✓	✓		35.48	↓2.25	57.58	47.95	↓1.62	50.23	52.52	↓1.84	50.23	48.83	↓4.05	20.99	↓2.00
✓		✓	34.87	↓2.86	55.13	46.16	↓3.74	49.46	48.74	↓4.12	44.34	46.72	↓8.05	22.70	↓0.29
✓	✓	✓	**33.35**	**↓4.38**	**52.37**	**45.31**	**↓5.55**	**47.01**	**47.95**	**↓5.74**	**43.74**	**46.66**	**↓8.38**	**20.66**	**↓2.33**



(a) Analysis of MRA placement (b) Analysis of Loss Weight λ (c) Analysis of the Training speed (d) Types of Canonical-View

Figure 5: Detailed analysis of the designed CanonSLU for CSLR on the MVSign-PT14, where (a), (c), (d) report the performance on the test set. (a) Hierarchical Placement Analysis of the MRA Module. (b) Sensitivity analysis of the distillation weight λ. (c) Comparison of canonical-view and non-canonical-view guidance. (d) Comparison of different canonical-view anchors.

MVSign-H2S. CanonSLU ranks first across all metrics, achieving BLEU-4/ROUGE scores of 6.19/20.71 on the dev set and 6.62/20.90 on the test set. Compared with the SpaMo, CanonSLU improves BLEU-4 by **1.95** and **2.29** and ROUGE by **2.88** and **2.91** on both sets, respectively. These results demonstrate its effectiveness on real-world multi-view data.

4.4 Ablation Study

Effectiveness of the CKT and MRA. We evaluate the individual contributions of the proposed modules on MVSign-PT14 across different viewpoints, as summarized in Table 4. Here, "Avg" denotes the mean WER of two views in each category, and Δ_{Avg} denotes the improvement relative to the baseline. The baseline model degrades significantly under large viewpoint variations, particularly for pitch views (*D30°*, *U30°*) and extreme horizontal angles (*R90°*, *L60°*). Introducing **CKT** reduces the overall WER by **2.25**, with larger gains on challenging views, including **1.62** for large-angle views and **4.05** for pitch views. This indicates that canonical-view supervision effectively alleviates projection-induced semantic ambiguity. Applying **MRA** alone yields a larger overall gain of **2.86** and improves every viewpoint, indicating that the proposed MRA configuration benefits multi-view recognition. Combining both modules achieves the best overall WER of **33.35%**. The improvement is especially pronounced under extreme viewpoint changes, where pitch views achieve **8.38** reduction. These results indicate that CKT and MRA are complementary, addressing cross-view discrepancy at the semantic and representation levels, respectively. Although CKT is applied only to non-frontal samples, its gradients update the model parameters shared across views, which may also improve frontal-view performance.

Mid-to-high layers are optimal for MRA. We analyze the impact of inserting MRA at different encoder stages, as shown in Fig. 5(a). The `layer1` setting is omitted due to prohibitive memory cost. Among single-layer configurations, `layer3` achieves the best performance, indicating that mid-to-high-level features are more suitable for modeling stable temporal relations. Extending MRA to `layer3+4` further improves non-frontal viewpoints performance, suggesting that motion representations from `layer3` and semantic representations from `layer4` are complementary. Considering both performance and computational cost, we adopt `layer3+4` in the final model.

Moderate λ gives the best trade-off. To evaluate the robustness of the proposed CKT, we conduct a sensitivity analysis on the distillation weight λ, varying it over $\{5, 10, 20, 40, 80\}$ (Figure 5(b)). The dev curve exhibits a relatively stable descent, reaching its optimal performance of 33.27% at $\lambda = 40$. When $\lambda = 80$, the performance slightly degrades, suggesting that an excessively large distillation weight may overwhelm the hard-label CTC constraints and cause the model to overfit the canonical-view. While the test set demonstrates more fluctuation—a common phenomenon under complex

multi-view settings—it perfectly aligns with the dev set at the optimal threshold, yielding the lowest test WER of 33.35% at $\lambda = 40$. Consequently, we set $\lambda = 40$ as the default configuration, ensuring an optimal equilibrium between task-specific learning and cross-view semantic transfer.

Canonical-view guidance improves optimization. To assess the effect of canonical-view prior guidance, we compare training with and without it. As shown in Figure 5(c), the guidance accelerates convergence, reaching the same WER earlier ($t_2 > t_1$), while maintaining consistently lower error rates throughout training. This suggests that the canonical view provides more stable supervision, improving both optimization efficiency and final recognition performance.

The frontal view is the most effective canonical anchor. We further analyze which view is the most suitable choice as the canonical anchor: *Front*, *L60°*, *R45°*, and *D30°*. All canonical-anchor models are trained using the same number of samples. The results are summarized in Figure 5(d), we report the WER on the **All**, the **Front**, and the average WER over **All side-view**. Among all views, using the frontal view as the canonical anchor yields the lowest WER (*e.g.* 33.35% (All), 47.17% (All side-view) and 20.66% (Front)). These results suggest that the frontal view serves as the most effective canonical anchor, as it provides the most complete and stable visual knowledge.

Performance-complexity analysis. Data generation is used only for offline dataset construction and does not affect inference. All experiments are conducted on the same server under the same GPU, batch-size, input-resolution, numerical-precision, and evaluation settings. As shown in Table 5, CanonSLU achieves the lowest WER with competitive complexity, outperforming CorrNet in inference speed and SignGraph in parameter count, FLOPs, and speed, thereby offering a favorable accuracy–efficiency trade-off.

Table 5: Accuracy-efficiency comparison on MVSign-PT14.

Methods	WER↓	Params(M)↓	FLOPs(G)↓	Inference Speed(videos/s)↑
VAC	39.57	34.30	**417.57**	**44.51**
SEN	38.17	34.51	423.15	26.01
CorrNet	34.80	**33.37**	427.97	26.03
SignGraph	35.13	36.63	491.12	17.30
CanonSLU	**33.35**	35.35	420.83	36.35

Visualization of Recognition Results. As shown in Figure 6, we further present a qualitative comparison of different methods. Under challenging viewpoint conditions, existing methods are more prone to semantic confusion and missing glosses. For example, in case (a), CorrNet misrecognizes "KOENNEN" as "MOEGLICH", while SEN further confuses "WEST" with "SUEDWEST"; in case (b), both CorrNet and SEN produce more obvious word substitution and semantic deviation errors. In contrast, our method correctly recovers the complete gloss sequences, demonstrating stronger robustness under multi-view settings. This indicates that CKT effectively alleviates cross-view semantic discrepancy, while MRA further refines stable motion cues.



Figure 6: Qualitative comparison of different methods. Red words indicate incorrect predictions, while blue words denote missing glosses.

5 Conclusions

In this paper, we introduce **MVSign**, a multi-view SLU benchmark spanning three sign languages and seven viewpoints, and propose **CanonSLU**, a canonical-view-guided baseline for multi-view SLU. MVSign combines naturally captured recordings with additional views synthesized through 3D whole-body motion reconstruction and target-view rendering, while preserving the source splits and annotations. CanonSLU treats the frontal view as a semantic anchor, combining CKT to transfer canonical knowledge to non-frontal views with MRA to strengthen motion-aware temporal representations under viewpoint changes. Experiments show that CanonSLU improves non-frontal performance, validating canonical-view guidance. As a controlled benchmark, MVSign still requires future validation on real multi-camera data and may introduce a synthetic-to-real gap.

AI USE STATEMENT

Large language models (LLMs) are used solely to aid and polish the writing of this manuscript, including grammar, clarity, and technical terminology. They are not used for literature retrieval, research ideation or execution, drafting paper sections, data generation, mathematical proof, or result interpretation. The authors reviewed and take full responsibility for all final text and scientific content. As required by ICLR policy, we confirm that LLMs are not listed as authors, and we take full responsibility for all content under our names.

ETHICS STATEMENT

This work uses human sign-language videos from publicly released benchmark datasets (*e.g.*, PHOENIX-2014T, CSL-Daily, How2Sign). These videos may contain personally identifiable and biometric information, including the appearance and faces of signers. We do not collect new human data or attempt to identify participants. We use the datasets according to their respective licenses and the consent and usage requirements documented by the dataset creators. We have reviewed the ICLR Code of Ethics and found no additional ethical concerns specific to this study. No potential conflicts of interest exist among the authors.

REPRODUCIBILITY STATEMENT

To ensure reproducibility, we provide the following resources: All implementation details, including network architectures, hyperparameters, and training protocols, are described in Appendix C.3. The CanonSLU implementation, data-generation scripts, and dataset metadata will be released upon publication. Random seeds are fixed for all reported experiments.

REFERENCES

Samuel Albanie, Gül Varol, Liliane Momeni, Hannah Bull, Triantafyllos Afouras, Himel Chowdhury, Neil Fox, Bencie Woll, Rob Cooper, Andrew McParland, et al. Bbc-oxford british sign language dataset. *arXiv preprint arXiv:2111.03635*, 2021.

Zhongang Cai, Wanqi Yin, Ailing Zeng, Chen Wei, Qingping Sun, Wang Yanjun, Hui En Pang, Haiyi Mei, Mingyuan Zhang, Lei Zhang, et al. Smpler-x: Scaling up expressive human pose and shape estimation. *Advances in Neural Information Processing Systems*, 36:11454–11468, 2023.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. Neural sign language translation. In *Computer Vision and Pattern Recognition*, pp. 7784–7793, 2018.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Computer Vision and Pattern Recognition*, pp. 10023–10033, 2020.

Yutong Chen, Fangyun Wei, Xiao Sun, Zhirong Wu, and Stephen Lin. A simple multi-modality transfer learning baseline for sign language translation. In *Computer Vision and Pattern Recognition*, pp. 5120–5130, 2022a.

Yutong Chen, Ronglai Zuo, Fangyun Wei, Yu Wu, Shujie Liu, and Brian Mak. Two-stream network for sign language recognition and translation. *Advances in Neural Information Processing Systems*, 35:17043–17056, 2022b.

Ka Leong Cheng, Zhaoyang Yang, Qifeng Chen, and Yu-Wing Tai. Fully convolutional networks for continuous sign language recognition. In *European Conference on Computer Vision*, pp. 697–714. Springer, 2020.

Aashaka Desai, Lauren Berger, Fyodor Minakov, Nessa Milano, Chinmay Singh, Kriston Pumphrey, Richard Ladner, Hal Daumé III, Alex X Lu, Naomi Caselli, et al. Asl citizen: a community-sourced dataset for advancing isolated sign language recognition. *Advances in Neural Information Processing Systems*, 36:76893–76907, 2023.

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. Sign language recognition: A large-scale multi-view dataset and comprehensive evaluation. In *Winter Conference on Applications of Computer Vision*, pp. 7887–7897. IEEE, 2025.

Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro-i Nieto. How2sign: a large-scale multimodal dataset for continuous american sign language. In *Computer Vision and Pattern Recognition*, pp. 2735–2744, 2021.

William T Freeman and Michal Roth. Orientation histograms for hand gesture recognition. In *International workshop on automatic face and gesture recognition*, volume 12, pp. 296–301. Zurich, Switzerland, 1995.

Shiwei Gan, Yafeng Yin, Zhiwei Jiang, Kang Xia, Lei Xie, and Sanglu Lu. Contrastive learning for sign language recognition and translation. In *International Joint Conference on Artificial Intelligence*, pp. 763–772, 2023.

Shiwei Gan, Yafeng Yin, Zhiwei Jiang, Hongkai Wen, Lei Xie, and Sanglu Lu. Signgraph: A sign sequence is worth graphs of nodes. In *Computer Vision and Pattern Recognition*, pp. 13470–13479, 2024.

Shiwei Gan, Yafeng Yin, Zhiwei Jiang, Lei Xie, Sanglu Lu, and Hongkai Wen. Mixsigngraph: A sign sequence is worth mixed graphs of nodes. *Advances in Neural Information Processing Systems*, 38:139371–139395, 2025.

Shiwei Gan, Xiao Liu, Yafeng Yin, Nan Liu, Kuizhuang Liu, Desibieer Tuerdaken, Zhiwei Jiang, Lei Xie, Sanglu Lu, and Hongkai Wen. Learning effective sign features without text for gloss-free sign language translation. In *Computer Vision and Pattern Recognition*, pp. 9827–9836, 2026.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *International Conference on Machine Learning*, pp. 369–376, 2006.

Mo Guan, Yan Wang, Guangkun Ma, Jiarui Liu, and Mingzu Sun. Mska: Multi-stream keypoint attention network for sign language recognition and translation. *Pattern Recognition*, 165:111602, 2025.

Shester Gueuwou, Sophie Siake, Colin Leong, and Mathias Müller. Jwsign: A highly multilingual corpus of bible translations for more diversity in sign language processing. *arXiv preprint arXiv:2311.10174*, 2023.

Dan Guo, Shengeng Tang, and Meng Wang. Connectionist temporal modeling of video and language: A joint model for translation and sign labeling. In *International Joint Conference on Artificial Intelligence*, pp. 751–757, 2019.

Jianyuan Guo, Peike Li, and Trevor Cohn. Bridging sign and spoken languages: Pseudo gloss generation for sign language translation. *Advances in Neural Information Processing Systems*, 38:77471–77499, 2025.

Junwei Han, George Awad, and Alistair Sutherland. Modelling and segmenting subunits for sign language recognition based on hand motion analysis. *Pattern recognition letters*, 30(6):623–633, 2009.

Aiming Hao, Yuecong Min, and Xilin Chen. Self-mutual distillation learning for continuous sign language recognition. In *International Conference on Computer Vision*, pp. 11303–11312, 2021.

Hezhen Hu, Weichao Zhao, Wengang Zhou, and Houqiang Li. Signbert+: Hand-model-aware self-supervised pre-training for sign language understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):11221–11239, 2023a.

Lianyu Hu, Liqing Gao, Zekang Liu, and Wei Feng. Continuous sign language recognition with correlation network. In *Computer Vision and Pattern Recognition*, pp. 2529–2539, 2023b.

Lianyu Hu, Liqing Gao, Zekang Liu, and Wei Feng. Self-emphasizing network for continuous sign language recognition. In *AAAI Conference on Artificial Intelligence*, volume 37, pp. 854–862, 2023c.

Jie Huang, Wengang Zhou, Qilin Zhang, Houqiang Li, and Weiping Li. Video-based sign language recognition without temporal segmentation. In *AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Eui Jun Hwang, Sukmin Cho, Junmyeong Lee, and Jong C Park. An efficient gloss-free sign language translation using spatial configurations and motion dynamics with llms. In *Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3901–3920, 2025.

Youngjoon Jang, Haran Raajesh, Liliane Momeni, Gül Varol, and Andrew Zisserman. Lost in translation, found in context: Sign language translation with contextual cues. In *Computer Vision and Pattern Recognition*, pp. 8742–8752, 2025.

Songyao Jiang, Bin Sun, Lichen Wang, Yue Bai, Kunpeng Li, and Yun Fu. Skeleton aware multimodal sign language recognition. In *Computer Vision and Pattern Recognition*, pp. 3413–3423, 2021.

Peiqi Jiao, Yuecong Min, Yanan Li, Xiaotao Wang, Lei Lei, and Xilin Chen. Cosign: Exploring co-occurrence signals in skeleton-based continuous sign language recognition. In *International Conference on Computer Vision*, pp. 20676–20686, 2023.

Peiqi Jiao, Yuecong Min, and Xilin Chen. Visual alignment pre-training for sign language translation. In *European Conference on Computer Vision*, pp. 349–367. Springer, 2024.

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

Siyuan Jing, Guangxue Wang, Haoyang Zhai, Qin Tao, Jun Yang, Bing Wang, and Peng Jin. Dual-view spatio-temporal feature fusion with cnn-transformer hybrid network for chinese isolated sign language recognition. *arXiv preprint arXiv:2506.06966*, 2025.

Jungeun Kim, Hyeongwoo Jeon, Jongseong Bae, and Ha Young Kim. Leveraging the power of mllms for gloss-free sign language translation. In *International Conference on Computer Vision*, pp. 21048–21058, 2025.

Oscar Koller, Jens Forster, and Hermann Ney. Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. *Computer Vision and Image Understanding*, 141:108–125, 2015.

Oscar Koller, Sepehr Zargaran, and Hermann Ney. Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms. In *Computer Vision and Pattern Recognition*, pp. 4297–4305, 2017.

Oscar Koller, Necati Cihan Camgoz, Hermann Ney, and Richard Bowden. Weakly supervised learning with multi-stream cnn-lstm-hmms to discover sequential parallelism in sign language videos. *IEEE transactions on pattern analysis and machine intelligence*, 42(9):2306–2320, 2019.

Chang Hao Lai, Rui Zhao, Xuewen Zhong, Jinsong Su, and Yidong Chen. Selective contrastive learning for gloss free sign language translation. In *Association for Computational Linguistics*, pp. 45648–45660, 2026.

Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *winter conference on applications of computer vision*, pp. 1459–1469, 2020.

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

Jing Lin, Ailing Zeng, Haoqian Wang, Lei Zhang, and Yu Li. One-stage 3d whole-body mesh recovery with component aware transformer. In *Computer Vision and Pattern Recognition*, pp. 21159–21168, 2023.

Yuecong Min, Aiming Hao, Xiujuan Chai, and Xilin Chen. Visual alignment constraint for continuous sign language recognition. In *International Conference on Computer Vision*, pp. 11542–11551, 2021.

Gyeongsik Moon, Shoou-I Yu, He Wen, Takaaki Shiratori, and Kyoung Mu Lee. Interhand2.6m: A dataset and baseline for 3d interacting hand pose estimation from a single rgb image. In *European Conference on Computer Vision*, pp. 548–564. Springer, 2020.

Muxin Pu, Mei Kuan Lim, and Chun Yong Chong. Siformer: Feature-isolated transformer for efficient skeleton-based sign language recognition. In *ACM International Conference on Multimedia*, pp. 9387–9396, 2024.

Xin Shen, Heming Du, Hongwei Sheng, Shuyun Wang, Hui Chen, Huiqiang Chen, Zhuojie Wu, Xiaobiao Du, Jiaying Ying, Ruihan Lu, et al. Mm-wlauslan: Multi-view multi-modal word-level australian sign language recognition dataset. *Advances in Neural Information Processing Systems*, 37:69700–69715, 2024.

Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28, 2015.

Shengeng Tang, Dan Guo, Richang Hong, and Meng Wang. Graph-based multimodal sequential embedding for sign language translation. *IEEE Transactions on Multimedia*, 24:4433–4445, 2021.

Shengeng Tang, Jiayi He, Dan Guo, Yanyan Wei, Feng Li, and Richang Hong. Sign-idd: Iconicity disentangled diffusion for sign language production. In *AAAI Conference on Artificial Intelligence*, volume 39, pp. 7266–7274, 2025a.

Shengeng Tang, Feng Xue, Jingjing Wu, Shuo Wang, and Richang Hong. Gloss-driven conditional diffusion models for sign language production. *ACM Transactions on Multimedia Computing, Communications and Applications*, 21(4):1–17, 2025b.

Garrett Tanzer and Biao Zhang. Youtube-sl-25: A large-scale, open-domain multilingual sign language parallel corpus. *arXiv preprint arXiv:2407.11144*, 2024.

Shuyuan Tu, Zhen Xing, Xintong Han, Zhi-Qi Cheng, Qi Dai, Chong Luo, and Zuxuan Wu. Stableanimator: High-quality identity-preserving human image animation. In *Computer Vision and Pattern Recognition*, pp. 21096–21106, 2025.

Dave Uthus, Garrett Tanzer, and Manfred Georg. Youtube-asl: A large-scale, open-domain american sign language-english parallel corpus. *Advances in Neural Information Processing Systems*, 36: 29029–29047, 2023.

Timo Von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision*, pp. 601–617, 2018.

Ching-Chen Wang, Ching-Te Chiu, Chao-Tsung Huang, Yu-Chun Ding, and Li-Wei Wang. Fast and accurate embedded dcnn for rgb-d based sign language recognition. In *International Conference on Acoustics, Speech and Signal Processing*, pp. 1568–1572. IEEE, 2020.

Fei Wang, Kun Li, Yiqi Nie, Zhangling Duan, Peng Zou, Zhiliang Wu, Yuwei Wang, and Yanyan Wei. Exploiting ensemble learning for cross-view isolated sign language recognition. In *Companion Proceedings of the ACM on Web Conference*, pp. 2453–2457, 2025a.

Xu Wang, Shengeng Tang, Peipei Song, Shuo Wang, Dan Guo, and Richang Hong. Linguistics-vision monotonic consistent network for sign language production. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1–5. IEEE, 2025b.

Ryan Wong, Necati Cihan Camgoz, and Richard Bowden. Sign2gpt: Leveraging large language models for gloss-free sign language translation. In *International Conference on Learning Representations*, 2024.

Min Xu, Sheng Liu, Yuan Feng, Yiheng Yu, Zhelun Jin, and Xuhua Yang. Hierarchical spatial-temporal enhancement network for continuous sign language recognition. In *International Conference on Acoustics, Speech and Signal Processing*, pp. 1–5. IEEE, 2025.

Yifan Yang, Yuecong Min, and Xilin Chen. S2net: Skeleton-aware slowfast network for efficient sign language recognition. In *Asian Conference on Computer Vision*, pp. 319–336, 2024.

Zhendong Yang, Ailing Zeng, Chun Yuan, and Yu Li. Effective whole-body pose estimation with two-stages distillation. In *International Conference on Computer Vision*, pp. 4210–4220, 2023.

Aoxiong Yin, Tianyun Zhong, Li Tang, Weike Jin, Tao Jin, and Zhou Zhao. Gloss attention for gloss-free sign language translation. In *Computer Vision and Pattern Recognition*, pp. 2551–2562, 2023.

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

Biao Zhang, Garrett Tanzer, and Orhan Firat. Scaling sign language translation. *Advances in neural information processing systems*, 37:114018–114047, 2024.

Rui Zhao, Liang Zhang, Biao Fu, Cong Hu, Jinsong Su, and Yidong Chen. Conditional variational autoencoder for sign language translation with cross-modal alignment. In *AAAI Conference on Artificial Intelligence*, volume 38, pp. 19643–19651, 2024.

Benjia Zhou, Zhigang Chen, Albert Clapés, Jun Wan, Yanyan Liang, Sergio Escalera, Zhen Lei, and Du Zhang. Gloss-free sign language translation: Improving from visual-language pretraining. In *International Conference on Computer Vision*, pp. 20871–20881, 2023.

Hao Zhou, Wengang Zhou, Yun Zhou, and Houqiang Li. Spatial-temporal multi-cue network for continuous sign language recognition. In *AAAI Conference on Artificial Intelligence*, volume 34, pp. 13009–13016, 2020.

Hao Zhou, Wengang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. Improving sign language translation with monolingual data by sign back-translation. In *Computer Vision and Pattern Recognition*, pp. 1316–1325, 2021.

Ronglai Zuo and Brian Mak. C2slr: Consistency-enhanced continuous sign language recognition. In *Computer Vision and Pattern Recognition*, pp. 5131–5140, 2022.

Ronglai Zuo, Fangyun Wei, and Brian Mak. Towards online continuous sign language recognition and translation. In *Empirical Methods in Natural Language Processing*, pp. 11050–11067, 2024.

SUPPLEMENTARY MATERIAL

Contents

A. Related Work 15
B. Detailed Implementation of Motion Relation Aggregation 16
C. Experimental Settings 17
D. Additional Experimental Results 19

A RELATED WORK

A.1 SIGN LANGUAGE UNDERSTANDING

CSLR. Early approaches rely on handcrafted features (Freeman & Roth, 1995; Han et al., 2009) and HMM-based temporal modeling (Koller et al., 2015; 2017). Deep learning methods adopt CNN- and RNN-based architectures (Koller et al., 2019; Guo et al., 2019; Hao et al., 2021; Min et al., 2021), typically combining 2D or 3D CNN backbones with temporal modules such as LSTMs (Shi et al., 2015). The CTC loss enables end-to-end training without frame-level gloss annotations (Graves et al., 2006; Hu et al., 2023b; Gan et al., 2024; Hu et al., 2023c). Recent studies incorporate auxiliary cues, including skeletons (Zhou et al., 2020; Yang et al., 2024), local regions (Zuo & Mak, 2022; Jiao et al., 2023), and depth information (Jiang et al., 2021; Wang et al., 2020), while improving backbones, attention, contrastive learning, temporal modeling, and large-scale pre-training (Cheng et al., 2020; Gan et al., 2023; Li et al., 2025). Online CSLR extend recognition and translation to streaming inputs, addressing the latency and memory demands associated with processing complete videos (Zuo et al., 2024). However, fine-grained hand-motion modeling remains challenging under viewpoint changes that alter hand appearance, motion projection, and occlusion.

SLT. Early methods formulate SLT as sequence-to-sequence learning with intermediate gloss supervision (Camgoz et al., 2018). Later works unify gloss recognition and sentence generation (Camgoz et al., 2020) and improve modeling with multi-stream architectures (Chen et al., 2022b), graph and contextual modeling (Tang et al., 2021; Jang et al., 2025), and multimodal or visual pre-training (Chen et al., 2022a; Jiao et al., 2024). Recent work further investigates multilingual scaling and cross-modal transfer through joint pretraining on sign language translation and text translation tasks (Zhang et al., 2024). Gloss-free SLT avoids costly gloss annotations and explores visual-language pre-training (Zhou et al., 2023), sign representation learning (Yin et al., 2023; Gan et al., 2026), pseudo-gloss and contrastive learning (Gan et al., 2025; Guo et al., 2025; Lai et al., 2026), and large language models (Wong et al., 2024; Kim et al., 2025). However, most methods assume single or fixed views, leaving viewpoint robustness underexplored.

A.2 SIGN LANGUAGE DATASETS

Large-scale, high-quality datasets are essential for deep SLU. Existing benchmarks target different tasks: WLASL (Li et al., 2020), MM-WLAuslan (Shen et al., 2024), National CSL-DP (Jing et al., 2025), and Multi-VSL (Dinh et al., 2025) support ISLR, with some providing multi-view samples, while PHOENIX-2014T (Camgoz et al., 2018) and CSL-Daily (Zhou et al., 2021) support CSLR and SLT with gloss and sentence annotations. How2Sign (Duarte et al., 2021) provides multimodal, multiview continuous ASL videos with English transcripts but no gloss labels. Web-collected corpora, including BoBSL (Albanie et al., 2021), JWSign (Gueuwou et al., 2023), YouTube-ASL (Uthus et al., 2023), and YouTube-SL-25 (Tanzer & Zhang, 2024), further expand dataset scale. Nevertheless, multi-view resources for continuous SLU remain scarce, as most CSLR and SLT benchmarks use fixed views and existing multi-view datasets mainly target ISLR. Since real multi-view collection requires calibrated and synchronized cameras with aligned annotations, we construct semantically aligned multi-view data from single-view videos through 3D reconstruction and view rendering.

B Detailed Implementation of Motion Relation Aggregation

Given an intermediate feature map $A_{i,l}^{v} \in \mathbb{R}^{C_l \times T_i \times H_l \times W_l}$, we first compute the frame-wise difference representation:

$$M_{i,l}^{v}(t) = A_{i,l}^{v}(t) - A_{i,l}^{v}(t-1), \qquad t = 2, \ldots, T_i. \tag{17}$$

The first temporal position is initialized as $M_{i,l}^{v}(1) = M_{i,l}^{v}(2)$. The difference representation is then refined by a temporal convolution:

$$\widetilde{M}_{i,l}^{v} = \phi_m(M_{i,l}^{v}), \tag{18}$$

where $\phi_m(\cdot)$ consists of a $3 \times 1 \times 1$ temporal convolution, batch normalization, and ReLU activation. In parallel, the original feature is projected into a value representation:

$$V_{i,l}^{v} = \phi_v(A_{i,l}^{v}), \tag{19}$$

where $\phi_v(\cdot)$ is a $1 \times 1 \times 1$ convolution followed by batch normalization.

For batched aggregation, we omit the sample, stage, and view indices and write the feature shape as $B \times C \times T \times H \times W$, where B is the batch size. We flatten the spatial dimensions into $N = HW$ locations. For each valid temporal offset $\delta \in \mathcal{T}_{\text{off}} = \{1, 2, 4\}$ with $\delta < T$, we pair the earlier frames $1, \ldots, T-\delta$ with the later frames $1+\delta, \ldots, T$. The corresponding refined motion features are denoted by $\widetilde{M}^{-}$ and $\widetilde{M}^{+}$, and their value features by V^{-} and V^{+}. For each pair index $t = 1, \ldots, T-\delta$, channel normalization gives:

$$\overline{M}^{-}_{b,t,n} = \frac{\widetilde{M}^{-}_{b,:,t,n}}{\max\left(\left\|\widetilde{M}^{-}_{b,:,t,n}\right\|_2, \varepsilon\right)}, \qquad \overline{M}^{+}_{b,t,m} = \frac{\widetilde{M}^{+}_{b,:,t,m}}{\max\left(\left\|\widetilde{M}^{+}_{b,:,t,m}\right\|_2, \varepsilon\right)}. \tag{20}$$

Here, $\varepsilon > 0$ prevents division by zero, b indexes the batch, and $n, m \in \{1, \ldots, N\}$ index spatial locations in the earlier and later frames, respectively. The motion similarity is computed by cosine similarity:

$$S_{b,t,n,m} = \left(\overline{M}^{-}_{b,t,n}\right)^{\mathsf{T}} \overline{M}^{+}_{b,t,m}. \tag{21}$$

For each earlier-frame location n, we select the top-k later-frame locations according to $S_{b,t,n,m}$. Let $\mathcal{N}_k^{-\to+}(n)$ denote this neighborhood, with its dependence on b, t, and δ omitted for brevity. The selected similarity scores are normalized as:

$$w^{-\to+}_{b,t,n,m} = \frac{\exp(S_{b,t,n,m})}{\sum_{q \in \mathcal{N}_k^{-\to+}(n)} \exp(S_{b,t,n,q})}, \qquad m \in \mathcal{N}_k^{-\to+}(n). \tag{22}$$

The selected value features are aggregated as:

$$O^{-\to+}_{b,:,t,n} = \sum_{m \in \mathcal{N}_k^{-\to+}(n)} w^{-\to+}_{b,t,n,m} V^{+}_{b,:,t,m}. \tag{23}$$

The superscript $- \to +$ denotes matching an earlier-frame location against later-frame locations. The resulting message is received by the earlier-frame location.

For the reverse matching direction, we exchange the earlier and later frames. For each later-frame location m, we independently select the top-k earlier-frame locations using $S_{b,t,n,m}$ over n and apply softmax over the selected locations. This gives $\mathcal{N}_k^{+\to-}(m)$ and $w^{+\to-}_{b,t,m,n}$, respectively. The reverse aggregation is:

$$O^{+\to-}_{b,:,t,m} = \sum_{n \in \mathcal{N}_k^{+\to-}(m)} w^{+\to-}_{b,t,m,n} V^{-}_{b,:,t,n}. \tag{24}$$

This message is received by the later-frame location.

For each offset δ, the messages $O^{-\to+}$ and $O^{+\to-}$ are accumulated at temporal positions $1, \ldots, T-\delta$ and $1+\delta, \ldots, T$, respectively. Let c_t denote the number of messages received at temporal position t across all valid offsets. The accumulated feature at each position is divided by $\max(c_t, 1)$, with the

denominator broadcast over the batch, channel, and spatial dimensions. After restoring the spatial dimensions, the averaged feature is denoted by $O^v_{i,l}$.

The aggregated feature is transformed by an output temporal convolution:

$$\widetilde{O}^v_{i,l} = \phi_o(O^v_{i,l}), \tag{25}$$

where $\phi_o(\cdot)$ consists of a $3 \times 1 \times 1$ temporal convolution and batch normalization. The gate is computed from the original feature and the nonnegative refined motion feature, both in their original spatial shape:

$$G^v_{i,l} = \mathrm{sigmoid}\left(\phi_g\left([A^v_{i,l}; \widetilde{M}^v_{i,l}]\right)\right). \tag{26}$$

Here, $\phi_g(\cdot)$ is a $1 \times 1 \times 1$ convolution, and $[\,;\,]$ denotes channel-wise concatenation. The MRA output is:

$$Y^v_{i,l} = \widetilde{O}^v_{i,l} \odot G^v_{i,l}, \tag{27}$$

where $\odot$ denotes element-wise multiplication. It is injected into the encoder through the residual update:

$$\widehat{A}^v_{i,l} = A^v_{i,l} + \alpha_l Y^v_{i,l}, \tag{28}$$

where α_l is a learnable weight. We use $k = 24$ and $k = 12$ at stages 3 and 4, respectively. Algorithm 1 summarizes the aggregation and gating operations.

Computational complexity. For each valid temporal offset δ, dense similarity computation requires $O(B(T-\delta)CN^2)$ time and $O(B(T-\delta)N^2)$ storage for the similarity tensor. Aggregating the selected top-k value features requires $O(B(T-\delta)NkC)$ time. These operations are performed in both matching directions and repeated over the valid offsets, with additional overhead from convolutional projections and top-k selection. Since selection follows dense similarity computation, the matching cost remains quadratic in the number of spatial positions N. This motivates placing MRA at stages 3 and 4, where the feature maps are smaller.

C Experimental Settings

C.1 Datasets.

All experiments are conducted on **MVSign**, which consists of **MVSign-PT14**, **MVSign-CSL**, and **MVSign-H2S**. MVSign-PT14 and MVSign-CSL are derived from PHOENIX-2014T (Camgoz et al., 2018) and CSL-Daily (Zhou et al., 2021), respectively. For each subset, we retain all original videos as the *Front* view and independently sample 800, 100, and 100 clips from the original train, dev, and test splits for each of the six non-frontal viewpoints $\{\mathrm{R45}^\circ, \mathrm{R90}^\circ, \mathrm{L30}^\circ, \mathrm{L60}^\circ, \mathrm{U30}^\circ, \mathrm{D30}^\circ\}$. The sampled clips are converted into the corresponding target views using the construction process described in Section 2, yielding 1,000 generated videos per viewpoint and 6,000 additional videos per subset. Each generated video inherits the split assignment, gloss annotation, and sentence annotation of its source clip, and sampling is performed independently across viewpoints while keeping all generated variants within the split of their source clips. MVSign-H2S is derived from the naturally captured multi-view videos in How2Sign (Duarte et al., 2021). We retain all original frontal-view videos and native $\mathrm{R90}^\circ$ videos, and sample 100 clips from each original train, dev, and test split to synthesize each of the five additional viewpoints $\{\mathrm{R45}^\circ, \mathrm{L30}^\circ, \mathrm{L60}^\circ, \mathrm{U30}^\circ, \mathrm{D30}^\circ\}$. This produces 300 generated videos per viewpoint, with 100 videos assigned to each split, and each video inherits the split assignment and sentence annotation of its source clip. Since How2Sign does not provide gloss annotations, MVSign-H2S is evaluated only for gloss-free SLT, whereas MVSign-PT14 and MVSign-CSL are evaluated for CSLR, gloss-based SLT, and gloss-free SLT.

C.2 Evaluation Metric.

Following prior SLU works (Chen et al., 2022a; Tang et al., 2025a; Gan et al., 2024; Tang et al., 2025b; Wang et al., 2025b), we adopt the Word Error Rate (WER) metric for the CSLR task, and ROUGE (ROUGE-L) and BLEU-1-4 for SLT tasks. Hereafter, ROUGE refers to ROUGE-L.

Algorithm 1 Motion Relation Aggregation

Require: Feature map $A \in \mathbb{R}^{B\times C\times T\times H\times W}$ with $T \geq 2$, temporal offset set $\mathcal{T}_{\text{off}} = \{1, 2, 4\}$, and top-$k$ parameter k
Ensure: MRA output Y
Compute frame differences: $M_t \leftarrow A_t - A_{t-1}$ for $t = 2, \ldots, T$
Initialize the first temporal position: $M_1 \leftarrow M_2$
Refine motion features: $\widetilde{M} \leftarrow \phi_m(M)$
Compute value features: $V \leftarrow \phi_v(A)$
Flatten the spatial dimensions of $\widetilde{M}$ and V, with $N = HW$
Initialize aggregated messages: $O \leftarrow \mathbf{0} \in \mathbb{R}^{B\times C\times T\times N}$
Initialize per-position counts: $c \leftarrow \mathbf{0} \in \mathbb{R}^{T}$
for all $\delta \in \mathcal{T}_{\text{off}}$ **do**
    if $\delta < T$ **then**
        Form temporally offset pairs $(\widetilde{M}^-, \widetilde{M}^+)$ and (V^-, V^+)
        Normalize $\widetilde{M}^-$ and $\widetilde{M}^+$ along the channel dimension with denominator $\max(\|\cdot\|_2, \varepsilon)$
        Compute dense cosine similarities across each frame pair
        Independently select top-k matches and apply softmax in both matching directions
        Aggregate V^+ into the earlier positions $1, \ldots, T - \delta$
        Aggregate V^- into the later positions $1 + \delta, \ldots, T$
        Accumulate both sets of messages into O
        $c_{1:T-\delta} \leftarrow c_{1:T-\delta} + 1$
        $c_{1+\delta:T} \leftarrow c_{1+\delta:T} + 1$
    end if
end for
Normalize each temporal position: $O_{b,:,t,:} \leftarrow O_{b,:,t,:} / \max(c_t, 1)$
Reshape O and $\widetilde{M}$ to $\mathbb{R}^{B\times C\times T\times H\times W}$
Apply the output projection: $\widetilde{O} \leftarrow \phi_o(O)$
Compute the gate: $G \leftarrow \text{sigmoid}\big(\phi_g([A; \widetilde{M}])\big)$
Modulate the aggregated feature: $Y \leftarrow \widetilde{O} \odot G$
return Y

WER. WER measures the minimum number of word-level substitutions, deletions, and insertions required to transform the predicted gloss sequence into the reference sequence:

$$\text{WER} = \frac{S + D + I}{N}, \tag{29}$$

where S, D, and I denote the numbers of substitutions, deletions, and insertions, respectively, and N is the number of tokens in the reference gloss sequence. Lower WER indicates better recognition performance.

BLEU. BLEU-n measures the modified precision of matched n-grams between a generated sentence $\hat{S}$ and a reference sentence S. The modified n-gram precision is defined as

$$p_n = \frac{\sum_g \min\left\{\text{Count}_{\hat{S}}(g), \text{Count}_S(g)\right\}}{\sum_g \text{Count}_{\hat{S}}(g)}. \tag{30}$$

BLEU-4 is computed as

$$\text{BLEU-4} = \text{BP} \exp\left(\frac{1}{4}\sum_{n=1}^{4} \log p_n\right), \tag{31}$$

where BP denotes the brevity penalty:

$$\text{BP} = \begin{cases} 1, & c > r, \\ \exp\left(1 - \frac{r}{c}\right), & c \leq r, \end{cases} \tag{32}$$

with c and r denoting the lengths of the generated and reference sentences, respectively. BLEU-1, BLEU-2, and BLEU-3 are computed analogously using $n = 1$, $n = 2$, and $n = 3$. Higher BLEU scores indicate better translation quality.

ROUGE-L. ROUGE-L is based on the length of the longest common subsequence (LCS) between the generated sentence $\hat{S}$ and the reference sentence S. Let $L = \mathrm{LCS}(\hat{S}, S)$, $c = |\hat{S}|$, and $r = |S|$. The LCS-based precision and recall are defined as

$$P_{\mathrm{LCS}} = \frac{L}{c}, \qquad R_{\mathrm{LCS}} = \frac{L}{r}. \tag{33}$$

ROUGE-L is then computed as their weighted harmonic mean:

$$\text{ROUGE-L} = \frac{(1+\gamma^2) P_{\mathrm{LCS}} R_{\mathrm{LCS}}}{R_{\mathrm{LCS}} + \gamma^2 P_{\mathrm{LCS}}}, \tag{34}$$

where γ controls the relative importance of recall and precision. Higher ROUGE-L scores indicate better translation quality.

C.3 Implementation Details.

We adopt the same data preprocessing as in previous work (Gan et al., 2024; 2025) for CanonSLU. Specifically, we use ResNet-18 as the visual encoder. All input videos are resized to 224×224 pixels and sampled with a frame interval of one. The models are optimized using Adam with an initial learning rate of 1×10^{-4} and a weight decay of 1×10^{-4}. For SLT, the learning rates of the translation module and the visual embedding layer are set to 1×10^{-5} and 1×10^{-3}, respectively. The batch size is set to 2, and CanonSLU is trained for 50 epochs. We use a MultiStep learning-rate schedule with milestones at epochs 20 and 35, reducing the learning rate by a factor of 0.2 at each milestone. For CKT, the distillation temperature and loss weight are set to $T_d = 8$ and $\lambda = 40$, respectively. MRA is inserted after the third and fourth stages of the visual encoder. Its temporal offsets are set to $\delta \in \{1, 2, 4\}$, with $k = 24$ and $k = 12$ selected spatial positions at the third and fourth stages, respectively. All experiments are implemented using PyTorch Distributed Data Parallel on 8 NVIDIA RTX 4080S GPUs.

D Additional Experimental Results

Table 6: Ablation results of different module combinations for Gloss-based SLT on MVSign-PT14.

Methods			DEV					TEST				
Baseline	CKT	MRA	BLEU1↑	BLEU2↑	BLEU3↑	BLEU4↑	ROUGE↑	BLEU1↑	BLEU2↑	BLEU3↑	BLEU4↑	ROUGE↑
✓			41.76	29.69	21.94	17.98	41.67	40.23	26.29	19.92	15.52	39.85
✓	✓		43.28	30.16	22.90	18.18	43.61	42.47	29.24	21.91	17.19	41.66
✓		✓	44.69	31.94	24.56	19.73	45.12	43.56	30.88	23.78	18.86	44.76
✓	✓	✓	**46.66**	**33.68**	**26.11**	**21.19**	**46.89**	**44.54**	**31.98**	**24.53**	**19.73**	**45.47**

Effectiveness of CKT and MRA for Gloss-based SLT. Table 6 evaluates the contributions of CKT and MRA to gloss-based SLT on MVSign-PT14. Both components improve translation performance individually, increasing test BLEU-4 by 1.67 and 3.34 points over the baseline, respectively. Combining them achieves the best results across all reported metrics, reaching test BLEU-4 and ROUGE scores of 19.73 and 45.47, with gains of 4.21 and 5.62 points. These results suggest that canonical-view guidance during CSLR pretraining and motion-aware feature learning provide complementary benefits for downstream translation.

Table 7: Comparison of viewpoint-wise CSLR performance (WER↓) on the paired-source MVSign-PT14 test set.

Methods	All	R90°	L60°	R45°	L30°	D30°	U30°	Front
SMKD (Hao et al., 2021)	53.55	63.52	54.72	52.20	55.37	55.97	54.72	38.36
VAC (Min et al., 2021)	51.12	62.87	55.36	50.94	52.83	54.09	50.94	30.82
SEN (Hu et al., 2023c)	50.40	58.49	53.46	50.91	50.94	54.08	53.46	31.45
CorrNet (Hu et al., 2023b)	46.06	55.09	47.91	47.82	48.23	45.13	47.03	31.19
SignGraph (Gan et al., 2024)	45.73	55.77	46.40	47.25	48.48	44.14	47.13	30.92
CanonSLU (Ours)	**44.65**	**53.46**	**45.71**	**45.28**	**47.84**	**43.35**	**46.17**	**30.79**

Paired-source viewpoint evaluation. Table 7 reports CSLR performance on the paired-source MVSign-PT14 test set. We randomly select 100 frontal-view clips from the original test split and generate all six non-frontal views for each clip, preserving their split assignments and annotations. CanonSLU achieves the lowest WER on the overall test set and every viewpoint. Its overall WER is 44.65%, improving over the strongest baseline, SignGraph (45.73%), by 1.08 points. The largest gains are observed at R45° and R90°, with reductions of 1.97 and 1.63 points over the strongest baseline for each view, respectively. CanonSLU also improves performance on the pitch views, reducing WER by 0.79 points at D30° and 0.86 points at U30°. The consistent improvements across views, including the frontal view, show that the method remains effective when source-clip content is controlled across viewpoints.

Table 8: Comparison of viewpoint-wise CSLR performance (WER↓) on MVSign.

Methods	MVSign-PT14 (Test set)								MVSign-CSL (Test set)							
	All	R90°	L60°	R45°	L30°	D30°	U30°	Front	All	R90°	L60°	R45°	L30°	D30°	U30°	Front
VAC (Min et al., 2021)	39.57	61.47	49.63	54.82	56.29	51.95	49.38	26.36	44.59	64.78	61.21	58.70	58.54	64.43	57.54	36.37
SEN (Hu et al., 2023c)	38.17	61.93	49.63	53.29	51.57	49.40	50.16	24.81	41.11	62.01	60.81	55.78	55.36	59.82	52.30	32.75
CorrNet (Hu et al., 2023b)	34.80	56.42	46.41	49.00	49.38	44.14	47.81	22.61	40.21	64.38	60.67	53.52	52.19	62.98	52.69	31.36
SignGraph (Gan et al., 2024)	35.13	57.92	48.13	49.62	49.33	48.74	48.10	22.95	37.63	57.96	56.69	53.20	51.27	58.60	48.21	29.95
CanonSLU (Ours)	**33.35**	**52.37**	**45.31**	**47.01**	**47.95**	**43.74**	**46.66**	**20.66**	**35.71**	**55.67**	**54.77**	**51.00**	**49.93**	**55.60**	**47.70**	**27.26**

View-wise CSLR Performance. Table 8 reports the test-set WER for each viewpoint on MVSign-PT14 and MVSign-CSL. CanonSLU achieves the lowest WER in every reported column, including the aggregate score, all six non-frontal views, and the frontal view. On MVSign-PT14, CanonSLU reduces the aggregate WER to 33.35%, improving over the best listed baseline, CorrNet (34.80%), by 1.45 points. The largest view-wise gain is observed at R90°, where the WER decreases from 56.42% to 52.37%. On MVSign-CSL, CanonSLU achieves an aggregate WER of 35.71%, which is 1.92 points lower than SignGraph (37.63%), and consistently improves performance across all viewpoints. The gains are particularly evident for D30° (3.00 points), R90° (2.29 points), and the frontal view (2.69 points). These results indicate that CanonSLU improves recognition across both horizontal and pitch variations rather than relying on a single viewpoint.

Table 9: Effect of temporal offset sets in MRA on MVSign-PT14.

$\mathcal{T}_{\text{off}}$	DEV			TEST		
	WER↓	BLEU-4↑	ROUGE↑	WER↓	BLEU-4↑	ROUGE↑
$\{1\}$	34.09	20.13	45.38	34.55	18.62	44.09
$\{1, 2\}$	34.99	19.96	45.54	35.35	18.37	44.12
$\{1, 2, 4\}$	**33.27**	**21.19**	**46.89**	**33.35**	**19.73**	**45.47**
$\{1, 2, 4, 8\}$	34.54	20.17	45.06	34.64	18.03	44.37

Selection of Temporal Offsets. Table 9 compares different temporal offset sets for MRA. The configuration $\mathcal{T}_{\text{off}} = \{1, 2, 4\}$ achieves the lowest WER and highest BLEU-4 and ROUGE on both splits, supporting its effectiveness for CSLR and gloss-based SLT. Performance does not improve monotonically as more offsets are included. Extending the set to $\{1, 2, 4, 8\}$ increases test WER from 33.35 to 34.64 and reduces test BLEU-4 from 19.73 to 18.03. These results support the default offset set $\{1, 2, 4\}$ for temporal relation aggregation.

Table 10: Effect of the top-k values of MRA at stages 3 and 4 on MVSign-PT14.

stage3	stage4	DEV			TEST		
		WER↓	BLEU-4↑	ROUGE↑	WER↓	BLEU-4↑	ROUGE↑
24	6	34.80	19.98	45.10	34.95	19.37	44.25
24	12	**33.27**	**21.19**	**46.89**	**33.35**	**19.73**	**45.47**
24	24	34.71	20.20	44.72	34.77	19.44	44.06
12	12	34.75	20.08	44.39	34.77	18.84	44.01
24	12	**33.27**	**21.19**	**46.89**	**33.35**	**19.73**	**45.47**
36	12	35.16	19.67	43.80	35.33	18.66	43.26

Table 11: Perceptual evaluation of synthesized videos. Scores range from 1 to 5, with higher values indicating better quality. Each source-generated pair is rated by five participants. The acceptance rate denotes the percentage of pairs whose mean rating across participants is at least 4.0 for each of the three criteria.

Subset	Views	Pairs	Visual realism	Temporal coherence	Motion consistency	Acceptance
MVSign-PT14	6 synthetic views	60 (10/view)	4.18	4.06	4.02	86.67%
MVSign-CSL	6 synthetic views	60 (10/view)	4.11	4.00	3.95	85.00%
MVSign-H2S	5 synthetic views	50 (10/view)	4.29	4.18	4.15	86.00%
Overall	–	170	4.19	4.07	4.03	85.88%

Effect of Top-k Selection. Table 10 examines the number of spatial positions selected at each MRA stage. We vary the top-k value at one stage while keeping the other fixed. Setting $k = 24$ at stage 3 and $k = 12$ at stage 4 achieves the best dev results across all three metrics, together with the lowest test WER and highest test ROUGE. The configuration $(12, 12)$ yields a slightly lower test BLEU-4 of 18.84, compared with 19.73 for $(24, 12)$. Overall, these results support $(24, 12)$ as the default configuration based on its validation performance and strong results across tasks.

Perceptual Evaluation. Table 11 reports the perceptual quality of synthesized videos, evaluated by five participants on 170 source-generated pairs, with 10 pairs sampled per synthetic viewpoint in each subset. The synthesized videos achieve overall scores of 4.19 for visual realism, 4.07 for temporal coherence, and 4.03 for motion consistency on a five-point scale. Moreover, 85.88% of the pairs receive average ratings of at least 4.0 on all three criteria, with acceptance rates ranging from 85.00% to 86.67% across the subsets. MVSign-H2S obtains the highest mean scores across all three criteria, while motion consistency receives the lowest overall rating, particularly on MVSign-CSL. These results suggest that the generated videos generally exhibit favorable **Visual realism**, **Temporal coherence**, and **Motion consistency** across viewpoints.

Table 12: CSLR qualitative results on MVSign-PT14 and MVSign-CSL. We use different colors to represent substitutions, deletions, and insertions, respectively.

Example (a)	**MVSign-PT14**
Groundtruth	IX KOENNEN GEFAHR GLATT BESONDERS MITTE REGION SUED DANN KALT
VAC	IX KOENNEN GEFAHR GLATT BESONDERS MITTE REGION SUED SONNE KALT
SignGraph	__ON__ IX KOENNEN GEFAHR GLATT BESONDERS MITTE REGION SUED DANN KALT
CanonSLU (Ours)	IX KOENNEN GEFAHR GLATT BESONDERS MITTE REGION SUED DANN KALT
Example (b)	**MVSign-PT14**
Groundtruth	OST REGION SONNE MEHR WARM MAXIMAL SECHS ZWANZIG SUED MAXIMAL SIEBEN ZWANZIG GRAD
VAC	OST REGION SONNE MEHR WARM MAXIMAL SECHS ZWANZIG SUED MAXIMAL SIEBEN ZWANZIG GRAD
SignGraph	OST REGION SONNE MEHR WARM MAXIMAL SECHS KOENNEN SUED MAXIMAL SIEBEN ZWANZIG GRAD
CanonSLU (Ours)	OST REGION SONNE MEHR WARM MAXIMAL SECHS ZWANZIG SUED MAXIMAL SIEBEN ZWANZIG GRAD
Example (c)	**MVSign-CSL**
Groundtruth	这 看 手机 这 女 他 你 女 朋友 是
VAC	这 看 手机 这 女 他 你 女 朋友 是
SignGraph	这 看 手机 这 女 他 多少 你 女 朋友 是
CanonSLU (Ours)	这 看 手机 这 女 他 你 女 朋友 是
Example (d)	**MVSign-CSL**
Groundtruth	小 张 这 女 是 你 公司 你 了解 他
VAC	小 张 这 女 是 他 公司 你 了解 他
SignGraph	小 张 这 女 是 你 公司 你 了解 他
CanonSLU (Ours)	小 张 这 女 是 你 公司 你 了解 他

Qualitative Results on CSLR. Table 12 compares recognition outputs on MVSign-PT14 and MVSign-CSL. VAC and SignGraph exhibit gloss substitutions, deletions, and insertions. For example, VAC omits "GEFAHR" and substitutes "DANN" with "SONNE" in example (a), while SignGraph replaces "ZWANZIG" with "KOENNEN" and misses "SIEBEN" in example (b). The MVSign-CSL examples similarly reveal pronoun confusion and omitted glosses. CanonSLU correctly recovers all four reference sequences, illustrating more accurate gloss recognition and better sequence completeness in these examples.

Table 13: SLT qualitative results on MVSign.

Example (a)	**MVSign-PT14**
Groundtruth	im verlauf kommt im norden und nordwesten teilweise kräftiger regen auf einzelne gewitter sind dabei
MMTLB	im verlauf kommt im norden und teilweise kräftiger regnet auf einzelne gewitter sind dabei
CV-SLT	im verlauf kommt im norden wolken nordwesten teilweise kräftiger sonntag auf einzelne gewitter sind dabei
CanonSLU (Ours)	im verlauf kommt im norden und nordwesten teilweise kräftiger regen auf einzelne gewitter sind dabei
Example (b)	**MVSign-CSL**
Groundtruth	我 给 你 预 订 了 一 个 生 日 蛋 糕 。
MMTLB	我 给 你 订 了 一 个 生 日 蛋 糕 在 超 市 。
CV-SLT	我 给 你 买 一 个 生 日 蛋 糕 。
CanonSLU (Ours)	我 给 你 预 订 了 一 个 生 日 蛋 糕 。
Example (c)	**MVSign-H2S**
Groundtruth	We're going to slide the front foot and drag the back foot whenever you're moving forward
GFSLT-VLP	We're going to slide the front foot and drag what good foot whenever you're moving forward
SpaMo	We're going to slide the stance foot and drag the back foot whenever you're came forward
CanonSLU (Ours)	We're going to slide the front foot and drag the back foot whenever you're moving forward

Qualitative Results on SLT. Table 13 presents translation examples from the three MVSign subsets. In example (a), MMTLB omits "nordwesten", while CV-SLT replaces "regen" with "sonntag", altering the weather description. In example (b), MMTLB adds an unsupported supermarket location, whereas CV-SLT changes the action from reserving a cake to buying one. In example (c), GFSLT-VLP mistranslates the back-foot reference, while SpaMo changes the front-foot reference and produces an ungrammatical description of forward movement. CanonSLU matches the reference translations in all three examples, preserving the geographic information, action meanings, and spatial relations.