

OLMD: Orientation-aware Long-term Motion Decoupling for Continuous Sign Language Recognition

Yiheng Yu, Sheng Liu*, Yuan Feng, Min Xu, Zhelun Jin, Xuhua Yang

Zhejiang University of Technology
{yhyu, edliu, fengyuan, mxu2019, zljin, xhyang}@zjut.edu.cn

Abstract

The primary challenge in continuous sign language recognition (CSLR) mainly stems from the presence of multi-orientational and long-term motions. However, current research overlooks these crucial aspects, significantly impacting accuracy. To tackle these issues, we propose a novel CSLR framework: **Orientation-aware Long-term Motion Decoupling (OLMD)**, which efficiently aggregates long-term motions and decouples multi-orientational signals into easily interpretable components. Specifically, our innovative Long-term Motion Aggregation (LMA) module filters out static redundancy while adaptively capturing abundant features of long-term motions. We further enhance orientation awareness by decoupling complex movements into horizontal and vertical components, allowing for motion purification in both orientations. Additionally, two coupling mechanisms are proposed: stage and cross-stage coupling, which together enrich multi-scale features and improve the generalization capabilities of the model. Experimentally, OLMD shows SOTA performance on three large-scale datasets: PHOENIX14, PHOENIX14-T, and CSL-Daily. Notably, we improve the word error rate (WER) on PHOENIX14 by an absolute **1.6%** compared to the previous SOTA.

Introduction

Continuous sign language features a distinctive grammatical structure and vocabulary (Jiao et al. 2023; Reagan 2007), making it challenging for hearing individuals without prior knowledge to learn. Lately, various research has been devoted to CSLR to promote smooth communication between deaf and hearing people (Jiao et al. 2023; Zuo and Mak 2022; Zheng et al. 2023; Min et al. 2021; Zhou et al. 2022), significantly narrowing the gap between them.

Recent CSLR research focuses on utilizing the motion inherent in sign language to improve recognition. For example, SEN (Hu et al. 2022a) enhances the discriminative of keyframes by exploiting motion-induced differences, while CorrNet (Hu et al. 2023a) calculates the correlation between adjacent frames to match areas with motion displacement. More recently, TCNet (Lu, Salah, and Poppe 2024) captures motion trajectory through its optical-flow-inspired module.

*Corresponding author

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

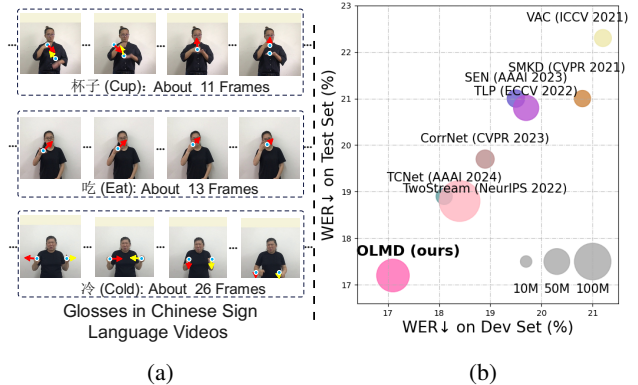


Figure 1: (a) We illustrate several common glosses with relevant video pieces from the CSL-Daily, highlighting the extensive presence of long-term and multi-orientational motions in sign language (yellow and red arrows indicate motion orientations), which traditional CSLR models struggle to manage. (b) OLMD Performance Comparison on the PHOENIX14: surpassing SOTA models by a large margin.

However, previous works ignore the long-term and multi-orientational nature of signs, which greatly affects the accuracy and robustness.

As shown in Fig. 1a, complex multi-orientational long-term motions are prevalent in sign language. Typically, the movements forming a sign span 8-13 frames (Buehler, Zisserman, and Everingham 2009). However, current CSLR methods primarily analyze only 1-2 adjacent frames (Hu et al. 2022a, 2023a; Gan et al. 2024), which is far from enough to capture a complete sign.

Additionally, much semantic information is embedded in orientations (Fenlon et al. 2013; Abdullahi and Chamnongthai 2022). To accurately decode this information, models should be orientation-aware. Unfortunately, conventional CSLR frameworks (Min et al. 2021; Hao, Min, and Chen 2021; Hu et al. 2023a) treat all orientations as a whole, resulting in confusion about orientational features and impacting accuracy.

To solve the above problems, we propose an **Orientation-aware Long-term Motion Decoupling (OLMD)**. Our approach includes a Long-term Motion Aggregation (LMA)

designed to capture long-term motion information. The LMA enables OLMD to focus on dynamic regions and adaptively aggregate long-term motions, significantly improving its ability to capture entire signs. For multi-orientational motions, we introduce an orientational motion decoupling strategy. This approach decouples motion features into their horizontal and vertical components, allowing for a clearer analysis of each orientation’s contribution. By enhancing orientation awareness through motion purification, the method ultimately improves the recognition of multi-orientational signs. Additionally, stage and cross-stage coupling complete the decoupling-coupling process, where the latter enhances robustness by utilizing multi-scale features.

Extensive experiments show our proposed **OLMD** outperforms all previous models on the three widely-used datasets: PHOENIX14 (Forster et al. 2015), PHOENIX14-T (Camgoz et al. 2018), and CSL-Daily (Zhou et al. 2021). Fig. 1b highlights the excellent performance of OLMD on PHOENIX14.

Our main contributions are summarized as follows:

- We innovatively design a Long-term Motion Aggregation (LMA) module, which effectively suppresses static redundancy while aggregating long-term motions in sign language.
- We effectively decouple sign motions and enhance orientation awareness through orientation-aware motion purification, improving the model’s understanding of complex signs.
- In extensive experiments, our OLMD demonstrates SOTA performance on three large datasets including PHOENIX14, PHOENIX14-T, and CSL-Daily.

Related Work

Continuous Sign Language Recognition

Continuous sign language recognition (CSLR) (Jiao et al. 2023; Zheng et al. 2023; Min et al. 2021; Zhou et al. 2022; Hu et al. 2022a, 2023a) aims to map untrimmed sign language videos to corresponding gloss¹ sequences in a weakly-supervised way. Early CSLR works use handcrafted features (Cooper and Bowden 2009; Yin, Chai, and Chen 2016) or HMM-based systems (Koller et al. 2016; Koller, Zargaran, and Ney 2017) to extract visual features and model long-time sequences. However, traditional handcrafted features often incur high computational costs and exhibit limited generalization performance. While HMMs struggle to capture nonlinear features and cannot be trained in an end-to-end manner. In recent years, deep learning methods have been extensively applied in CSLR, effectively addressing the limitations of traditional handcrafted features. Furthermore, the use of CTC loss (Graves et al. 2006) facilitates end-to-end training of models. Overall, deep learning and CTC loss serve as the foundation for contemporary CSLR models.

Regarding inputs, present CSLR models can be broadly categorized into *single-cue* and *multi-cue*. *Multi-cue* models, such as TwoStream (Chen et al. 2022), typically achieve

¹Gloss is the lexical unit of annotated continuous sign language.

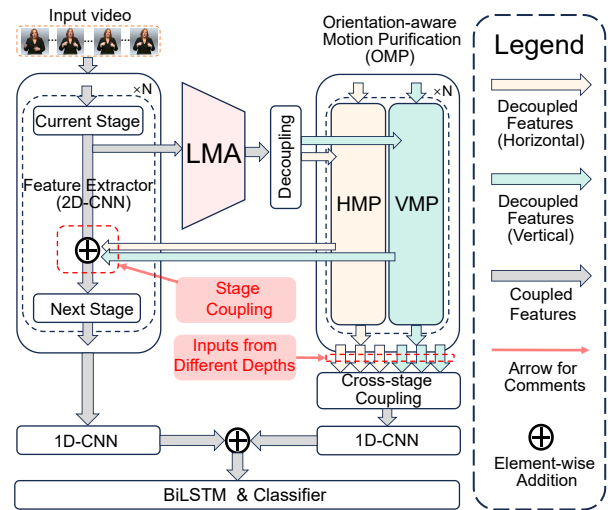


Figure 2: An overview of the proposed OLMD. After each stage of the Feature Extractor, frame-wise features first aggregate long-term motion information via LMA, which is then decoupled into horizontal and vertical components. HMP (Horizontal Motion Purification) and VMP (Vertical Motion Purification) are subsequently applied to enhance orientation-specific motion awareness. Stage and cross-stage coupling leverage enhanced features within and across stages, ensuring the integrity of decoupling-coupling while enriching the utilization of multi-scale features. Finally, two 1D-CNNs share architecture for downsampling and local temporal modeling, while the BiLSTM is used for global temporal modeling.

high accuracy by incorporating rich information. However, these methods also introduce higher computational complexity and deployment costs. Considering the dual requirements of CSLR for real-time and accuracy, the current research strives to find a balance point based on enhancing the *single-cue* models. For instance, VAC (Min et al. 2021) and SMKD (Hao, Min, and Chen 2021) introduce alignment losses to optimize the synergy between local and global features of the model. CVT-SLR (Zheng et al. 2023) and CTCA (Guo et al. 2023) utilize pre-trained linguistic knowledge to assist in the extraction of visual features. TLP (Hu et al. 2022b) refines the pooling layers, more intelligently extracting information across different levels of the temporal sequence. In general, these advances in *single-cue* CSLR have laid the foundation for the introduction of **OLMD**.

Motion Decoupling

Motion decoupling has been effectively utilized in various fields to improve model performance and reduce noise. For example, (Wang et al. 2024) employs a motion decoupling technique based on separating low-frequency and high-frequency signals, which reduces the noise interference in Video Motion Magnification (VMM). (Bian et al. 2023) proposes a decoupling architecture based on distinct objects, aimed at estimating the motion of tissues and instruments under various constraints in medical imaging. (He et al.

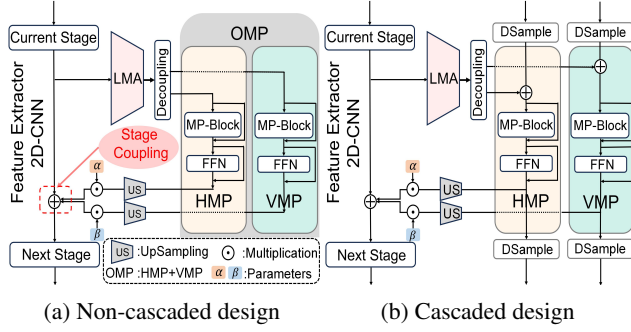


Figure 3: Details of the decoupling and stage-coupling are shown. (a) and (b) represent different designs of the OMP.

2024) proposes a decoupling architecture based on nonlinear Thin Plate Spline (TPS) transformation, capable of separating underlying motion features and guiding the synthesis of gesture videos. In this paper, we innovatively propose an orientational motion decoupling method. By decoupling complex motions into simple horizontal and vertical components, the model can enhance its orientation awareness through orientation-aware motion purification. As a result, our **OLMD** becomes better equipped to handle multi-orientational sign movements.

Methods

Overview

The purpose of CSLR is to convert a video with T frames $V = \{v_t\}_{t=1}^T \in \mathcal{R}^{T \times H \times W \times 3}$ into a series of glosses $\mathcal{G} = \{g_i\}_{i=1}^N$, where N represents the length of the gloss sequence. Traditional methods employ a 2D-CNN to extract frame-wise features $X = \{x_t\}_{t=1}^T \in \mathbb{R}^{T \times d}$. These features are then processed by a 1D-CNN and a BiLSTM for local and global temporal modeling. Finally, the outputs are classified while optimized by CTC loss.

Fig. 2 shows **OLMD**, an innovative CSLR framework. After each stage of the Feature Extractor, we employ LMA to aggregate long-term motion information, followed by orientational motion decoupling. The decoupled features are then purified and enhanced in the VMP and HMP, respectively. Stage coupling integrates features within each stage before forwarding them to the next, while cross-stage coupling combines decoupled features from different depths.

Decoupling-coupling Method

Conventional models typically capture all motions as a whole (Min et al. 2021; Hao, Min, and Chen 2021), which can result in feature confusion and reduce accuracy. In contrast, our method decouples motion features into their horizontal and vertical components, enhancing the model’s ability to discern complex patterns. Simply, the method consists of four parts: 1) motion decoupling, 2) orientation-aware motion purification, 3) stage coupling, and 4) cross-stage coupling.

Motion Decoupling. As shown in Fig. 3a, the frame-wise features $X \in \mathcal{R}^{C \times T \times H \times W}$ from the current stage first aggregate motion information to obtain $X_{agg} \in \mathcal{R}^{C \times T \times H \times W}$.

The aggregated X_{agg} encompasses multi-orientational motions. To simplify these complex movements, we draw from physics-based methods by projecting motion features into a lower-dimensional space. This process decouples intricate motions into simpler, unidirectional components, making the dynamics clearer and more analyzable. Specifically, to isolate vertical motion, we exclude all horizontal components, and vice versa. The process can be expressed as:

$$\begin{aligned} X_{agg} &= \text{Agg}_{(k)}(X), \\ X_h, X_v &= \mathcal{DP}(X_{agg}), \end{aligned} \quad (1)$$

where $\text{Agg}_{(k)}(\cdot)$ denotes the aggregation of long-term motion (detailed in the next section), and k represents the motion context length. $X_h \in \mathcal{R}^{C \times T \times W}$, $X_v \in \mathcal{R}^{C \times T \times H}$ denote the decoupled horizontal and vertical components, respectively. $\mathcal{DP}(\cdot)$ refers to a decoupling operation (e.g., Maxpool, Avgpool) identified in the ablation study.

Orientation-aware Motion Purification (OMP). OMP purifies the decoupled features, enhancing the model’s orientation awareness. As shown in Fig. 3a, OMP includes horizontal-aware motion purification (HMP) and vertical-aware motion purification (VMP) for purifying vertical and horizontal motions, respectively. In HMP and VMP, the decoupled features X_h and X_v are treated as specialized 2D images. The MP-Block (2D CNNs, detailed in the supplementary materials) is employed to refine motion information in specific orientations, followed by a residual connection:

$$X_{h/v} = \text{MP-Block}(X_{h/v}) + X_{h/v}, \quad (2)$$

where $X_{h/v} \in \mathcal{R}^{C \times T \times (W/H)}$ denotes X_h or X_v , encompassing only horizontal or vertical motions, respectively.

Subsequently, the FFN (Vaswani et al. 2017) is employed to apply nonlinear transformations to channels, enhancing the representational capacity of OMP:

$$\text{FFN}(X_{h/v}) = \max(0, \text{Norm}(X_{h/v}W_1))W_2, \quad (3)$$

where $W_1 \in \mathcal{R}^{C_{hid} \times C_{in} \times 1 \times 1}$, $W_2 \in \mathcal{R}^{C_{out} \times C_{hid} \times 1 \times 1}$ represent two 2D CNNs with a kernel size of 1, $\text{Norm}(\cdot)$ represents layer normalization.

Inspired by LGD (Qiu et al. 2019), we also design the cascaded OMP. As shown in Fig. 3b, decoupled features from the previous stage are downsampled and fused into the next, progressively enriching the information representation and increasing the network depth of OMP:

$$X_{h/v} = \mathcal{DS}(X_{h/v}^{pre}) + X_{h/v}, \quad (4)$$

where $X_{h/v}^{pre} \in \mathcal{R}^{C \times T \times (W/H)}$ denotes the decoupled feature from the previous stage. $\mathcal{DS}(\cdot)$ denotes the downsampling method, which employs a 2D CNN with a stride of 2 in the last dimension

Stage Coupling. The enhanced X_h, X_v are coupled to update the features X in the current stage of the Feature Extractor:

$$X = X + \alpha \times \text{US}(X_h) + \beta \times \text{US}(X_v), \quad (5)$$

where $\text{US}(\cdot)$ represents the up-sampling operation (we repeat the features at each position). α, β are learnable parameters used to control the contribution of orientations. They are initialized to zero to preserve the original features.

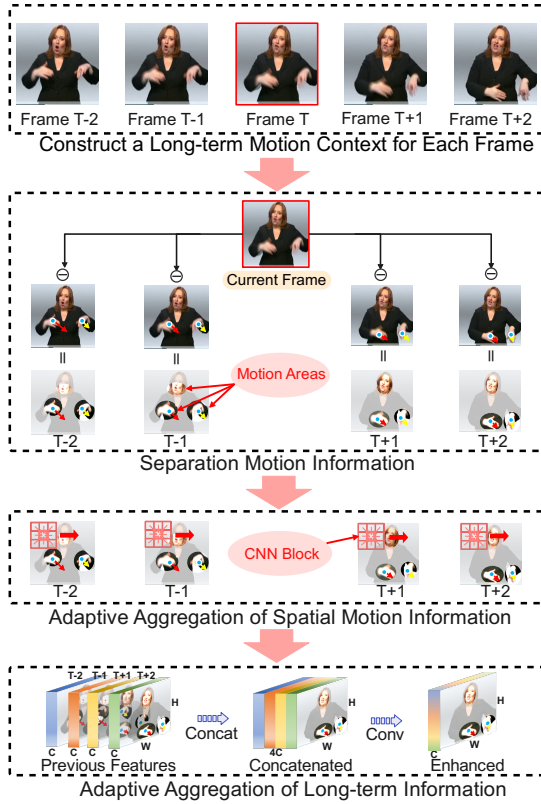


Figure 4: The main idea of our Long-term Motion Aggregation (LMA) module, illustrated with a context length of 5.

Cross-stage Coupling. Cross-stage coupling leverages multi-scale information by coupling features from different depths:

$$X_{CS} = \sum_{i=1}^n (W_i(\mathcal{P}_{dim=W}(X_h^i) + \mathcal{P}_{dim=H}(X_v^i))), \quad (6)$$

where n depends on the number of decoupling iterations. X_h^i and X_v^i are derived from the i -th decoupling. $\mathcal{P}_{dim=H/W}(\cdot)$ represents the global average pooling operation on height or width dimension. W_i denotes the i -th fully connected layer. $X_{CS} \in \mathcal{R}^{C \times T}$ denotes the features coupled through cross-stage coupling.

Long-term Motion Aggregation

Fig. 1a illustrates the frequent long-term motions observed in sign language, while current works only focus on adjacent 1-2 frames (Hu et al. 2022a, 2023a; Gan et al. 2024), which adversely impacts accuracy. To tackle the problem, we propose the LMA module, as illustrated in Fig. 4. Each component is detailed below.

The $X = [x_1, x_2, \dots, x_T] \in \mathcal{R}^{C \times T \times H \times W}$ represents the sequence of features for T frames, where each frame $x_i \in \mathcal{R}^{C \times H \times W}$. To reduce computational costs, the channel dimension is compressed to C/r by a CNN. Following this, we establish a long-term motion context for each frame through the following operation:

$$X_{(2n+1)} = \mathcal{S}_{(2n+1)}(X), \quad (7)$$

where $\mathcal{S}_{2n+1}(\cdot)$ denotes the sliding window operation with a window size of $2n + 1$, $X_{(2n+1)} \in \mathcal{R}^{(2n+1) \times C/r \times T \times H \times W}$ denotes the feature sequence with motion context $2n + 1$.

We represent $X_{(2n+1)}$ unfolded along the motion context as $[\mathcal{C}_{(-n)}, \dots, \mathcal{C}_{(0)}, \dots, \mathcal{C}_{(n)}]$, where $\mathcal{C}_{(i)} \in \mathcal{R}^{C/r \times T \times H \times W}$ denotes the i -th frames offset from the current frames. The motion areas are subsequently separated through the following calculations:

$$\mathcal{C}'_{(i)}(h, w) = \mathcal{C}_{(0)}(h, w) - \mathcal{C}_{(i)}(h, w), \quad i \in [-n, n], i \neq 0, \quad (8)$$

(h, w) denotes the spatial location, if the feature at this location remains unchanged (e.g., background areas), the result will tend toward 0. This indicates that the model can focus on processing the key areas without being affected by redundancies.

To aggregate the discrete spatial motion regions and enhance the discriminative ability of the features, a shared CNN block is applied to sequentially process each frame:

$$\mathcal{C}'_{(i)} = \text{ConvBlock}(\mathcal{C}'_{(i)}), \quad i \in [-n, n], i \neq 0, \quad (9)$$

where $\text{ConvBlock}(\cdot)$ mainly consists of three CNNs, each with a kernel size of $1 \times 3 \times 3$.

Subsequently, $\{\mathcal{C}'_{(-n)}, \dots, \mathcal{C}'_{(n)}\}$ are concatenated along the channel dimension, and a CNN is used to adaptively aggregate the long-term motion information while reducing the channels to C/r :

$$X' = \text{Conv}_{1 \times 1 \times 1}(\text{Concat}([\mathcal{C}'_{(-n)}, \dots, \mathcal{C}'_{(n)}])), \quad (10)$$

where $X' \in \mathcal{R}^{C/r \times T \times H \times W}$. Finally, we restore the channels back to C with another CNN and apply a residual connection to retain the original information:

$$X^{out} = \text{Conv}_{1 \times 1 \times 1}(X') + X. \quad (11)$$

Loss Design

The losses in OLMD are primarily divided into CTC losses and self-distillation losses.

CTC losses. CTC loss (Graves et al. 2006) is the most widely used loss in CSLR tasks. It employs dynamic programming to address the alignment problem between outputs and labels, enabling end-to-end training. In OLMD, the total CTC loss is represented as:

$$\mathcal{L}_{CTC} = \mathcal{L}_{CTC}^{local1} + \mathcal{L}_{CTC}^{local2} + \mathcal{L}_{CTC}^{global}, \quad (12)$$

where $\mathcal{L}_{CTC}^{local1}$ and $\mathcal{L}_{CTC}^{local2}$ represent supervision for the outputs of two 1D-CNNs, $\mathcal{L}_{CTC}^{global}$ represents supervision for the outputs of BiLSTM. Following SMKD (Hao, Min, and Chen 2021), we share all classifiers before computing CTC losses.

Self-distillation losses. Inspired by VAC (Min et al. 2021), the outputs of the entire network are treated as the teacher, while the outputs of two 1D-CNNs are considered the students. The loss is computed using KL divergence:

$$\mathcal{L}_{Distill} = \mathcal{L}_{KL}^{g2l1} + \mathcal{L}_{KL}^{g2l2}. \quad (13)$$

Methods	Cues	PHOENIX14				PHOENIX14-T	
		Dev(%)		Test(%)		Dev(%)	Test(%)
		del/ins	WER ↓	del/ins	WER ↓	WER ↓	WER ↓
SFL (Niu and Mak 2020)	video	7.9 / 6.5	26.2	7.5 / 6.3	26.8	25.1	26.1
VAC (Min et al. 2021)	video	7.9 / 2.5	21.2	8.4 / 2.6	22.3	21.4	23.9
SMKD (Hao, Min, and Chen 2021)	video	6.8 / 2.5	20.8	6.3 / 2.3	21.0	20.8	22.4
TLP (Hu et al. 2022b)	video	6.3 / 2.8	19.7	6.1 / 2.9	20.8	19.4	21.2
SEN (Hu et al. 2022a)	video	5.8 / 2.6	19.5	7.3 / 4.0	21.0	19.3	20.7
AdaBrowse+ (Hu et al. 2023b)	video	6.0 / 2.5	19.6	5.9 / 2.6	20.7	19.5	20.6
CorrNet (Hu et al. 2023a)	video	5.6 / 2.8	18.8	5.7 / 2.3	19.4	18.9	20.5
CoSign (Jiao et al. 2023)	keypoints	-	19.7	-	20.1	19.5	20.1
SignGraph (Gan et al. 2024)	video	4.9 / 2.0	18.2	4.9 / 2.0	19.1	17.8	19.1
TCNet (Lu, Salah, and Poppe 2024)	video	5.5 / 2.4	18.1	5.4 / 2.0	18.9	18.3	19.4
CNN+LSTM+HMM* (Koller et al. 2020)	video+mouth+hand	-	26.0	-	26.0	22.1	24.1
DNF* (Cui, Liu, and Zhang 2019)	video+optical flow	7.3 / 3.3	23.1	6.7 / 3.3	22.9	22.7	24.0
STMC* (Zhou et al. 2022)	video+hand+face+pose	7.7 / 3.4	21.1	7.4 / 2.6	20.7	19.6	21.0
C ² SLR* (Zuo and Mak 2022)	video+keypoints	-	20.5	-	20.4	20.2	20.4
TwoStream-SLR* (Chen et al. 2022)	video+keypoints	-	18.4	-	18.8	17.7	19.3
OLMD (ours)	video	4.8 / 2.4	17.1	4.7 / 2.3	17.2	17.1	18.4

Table 1: Performance comparison with the latest methods on PHOENIX14 and PHOENIX14-T, * indicates the introduction of additional visual cues.

Total loss. The total loss is a combination of the CTC and self-distillation losses:

$$\mathcal{L} = \gamma_1 \mathcal{L}_{CTC} + \gamma_2 \mathcal{L}_{Distill}, \quad (14)$$

where γ_1 and γ_2 are two hyperparameters used to balance contributions, we set them to 1 and 25, respectively.

Experiments

Datasets and Evaluation

In this paper, we mainly use three large CSLR datasets:

PHOENIX14 (Forster et al. 2015) is a popular CSLR dataset from German TV weather reports with 1,295 words, recorded by 9 signers. It includes 5,672 training, 540 development (Dev), and 629 testing (Test) videos.

PHOENIX14-T (Camgoz et al. 2018) is an expanded version of PHOENIX-2014, offering 7,096 training, 519 development (Dev), and 642 testing (Test) videos. It has a vocabulary of 1,115 sign language signs and 3,000 German words, serving sign recognition and translation tasks.

CSL-Daily (Zhou et al. 2021) is a large-scale Chinese sign language dataset filmed by 10 different signers, covering a variety of daily life themes with over 20,000 sentences. The dataset is split into 18,401 training samples, 1,077 development samples, and 1,176 test samples, featuring 2,000 sign language and 2,343 Chinese text vocabularies.

Evaluation Metric. We use word error rate (WER) as the performance metric for our model. It measures the minimum substitutions (#sub), insertions (#ins), and deletions (#del) required to match a predicted sentence with the reference (#ref):

$$\text{WER} = \frac{\#sub + \#ins + \#del}{\#ref}, \quad (15)$$

note that the **lower** WER, the **better** accuracy.

Implementation Details

This section primarily details the implementation of OLMD.

Network details. For Feature Extractor, we employ a ResNet34 (He et al. 2015), which is pretrained on ImageNet (Deng et al. 2009). For 1D-CNNs, we adopt SOTA configurations $\{K5, P2, K5, P2\}$, where $K5$ represents a convolution with a kernel size of 5, and $P2$ represents pooling with a kernel size of 2. We adopt the pooling method from TLP (Hu et al. 2022b), setting the hidden layers of the BiLSTM to 1024.

Training and test strategy. During training, we set the batch size to 2 and the initial learning rate to 0.001, reducing to 30% at epochs 25 and 40. We default to using the Adam optimizer with a weight decay of 0.001, iterating for a total of 70 epochs. All input frames are first resized to 256x256 and then randomly cropped to 224x224 during training, with a 50% chance of horizontal flipping and a 20% probability of temporal scale adjustment. For inference, we simply use a central crop of 224x224. Finally, all the training and testing are completed on 1 NVIDIA A6000 GPU.

Comparison with SOTA Methods

We verify the performance of OLMD on three large datasets.

On PHOENIX14 and PHOENIX14-T. Tab. 1 presents a comparison of OLMD with other SOTA models on PHOENIX14 and PHOENIX14-T datasets. All models are categorized into two groups: single-cue and multi-cue (denoted with *). The single-cue models only use RGB videos as input, while the multi-cue models use pre-extracted keypoints or heatmaps to aid recognition.

Notably, OLMD outperforms all single-cue and even multi-cue models. For instance, compared to the single-cue SOTA TCNet (Lu, Salah, and Poppe 2024), our WER on

Methods	Dev(%)	Test(%)
LS-HAN (Huang et al. 2018)	39.0	39.4
SEN (Hu et al. 2022a)	31.1	30.7
AdaBrowse+ (Hu et al. 2023b)	31.2	30.7
CorrNet (Hu et al. 2023a)	30.6	30.1
CoSign (Jiao et al. 2023)	28.1	27.2
TCNet (Lu, Salah, and Poppe 2024)	29.7	29.3
SignGraph (Gan et al. 2024)	27.3	26.4
TwoStream-SLR* (Chen et al. 2022)	25.4	25.3
OLMD (ours)	25.8	24.7

Table 2: Performance comparison with the latest methods on CSL-Daily, * indicates the introduction of additional visual cues.

PHOENIX14 is reduced by 1.0% (Dev) and 1.7% (Test). On PHOENIX14-T, it’s still reduced by 1.2% (Dev) and 1.0% (Test). The multi-cue SOTA Two-stream (Chen et al. 2022) relies on auxiliary keypoint sequences and a large network structure to enhance recognition capabilities. However, OLMD surpasses it with single-cue input and a lightweight architecture. Remarkably, on the PHOENIX14, OLMD lowers the WER by 1.3% (Dev) and 1.6% (Test). The above results highlight the effectiveness of OLMD in capturing and leveraging complex movements in sign language, leading to a significant enhancement in performance.

On CSL-Daily. CSL-Daily is a challenging dataset with the largest vocabulary and the most sign language videos among all publicly available CSLR datasets.

Tab. 2 shows our superior performance. Compared to the previous best single-cue method SignGraph (Gan et al. 2024), OLMD exhibited a reduction in WER of 1.5% (Dev) and 1.7% (Test). Besides, we even show a WER decline of 0.6% on the Test compared to the best multi-cue method TwoStream. The above results indicate the advancement of OLMD and its adaptability to larger datasets.

Ablation Studies

We evaluate the effectiveness of each component via ablation studies on PHOENIX14.

Ablations for the locations of decoupling. In Tab. 3, we ablate the decoupling positions after stages 2, 3, and 4. According to the results, decoupling at any position enhances performance, with more decoupling positions leading to more significant gains. Finally, OLMD applies decoupling after stages 2, 3, and 4, resulting in a 2.1% improvement on Dev and 2.3% on Test compared to the baseline.

Ablations for LMA, HMP and VMP. Tab. 4 presents an ablation study of three key components: LMA, HMP, and VMP. LMA aggregates long-term information, while HMP and VMP purify vertical and horizontal motions, respectively. When features are decoupled and purified without LMA, performance improvements are trivial (0.3% on Dev and 0.1% on Test). However, integrating LMA with either HMP or VMP results in significant gains (2% on Dev and 1.7% on Test). The combination of all three components achieves the highest performance boost, particularly on the

Stage 2	Stage 3	Stage 4	Dev (%)	Test (%)
-	-	-	19.2	19.5
✓	-	-	17.8	18.4
-	✓	-	17.5	18.5
-	-	✓	17.6	18.4
✓	✓	-	17.6	17.9
-	✓	✓	17.3	17.7
✓	✓	✓	17.1	17.2

Table 3: Ablations for the locations of decoupling on the PHOENIX14. Stage n indicates decoupling after the n-th stage.

LMA	HMP	VMP	Dev (%)	Test (%)
-	-	-	19.2	19.5
✓	-	-	18.3	19.3
-	✓	✓	18.9	19.4
✓	✓	-	17.2	17.9
✓	-	✓	17.2	17.8
✓	✓	✓	17.1	17.2

Table 4: Ablations for LMA, HMP and VMP on the PHOENIX14.

Test set, with an additional 0.7% improvement.

Ablations for LMA’s motion context length & parameters & FLOPs & inference time. Tab. 5 presents the effects of varying motion contexts on model performance, parameters, FLOPs, and inference time. The results indicate a consistent decrease in WER with increasing context, while other metrics show varying increases. At a context length of 9, WER achieves its minimum, with parameters at 80.74M, FLOPs at 474.9G, and inference time at 0.047s. This demonstrates that the model remains both lightweight and real-time, leading us to select a context length of 9 for LMA.

Ablations for two designs of OMP. In Tab. 6, we compare non-cascaded OMP with cascaded OMP. The results show that cascaded OMP performs better, particularly reducing WER by 0.9% on the Test. This indicates that cascaded OMP progressively enhances features, increases the OMP’s depth, and improves the model’s generalization.

Ablations for decoupling methods. In Tab. 7, we evaluate various decoupling strategies: No-decoupling, MaxPool, AvgPool, and MaxPool+AvgPool. For the No-decoupling scenario, we adapted the 2D CNNs from OMP to 3D CNNs. The results demonstrate that all decoupling methods enhance performance relative to the No-decoupling approach, with AvgPool achieving the highest model performance. Consequently, we adopt AvgPool as the decoupling strategy for OLMD.

Ablations for stage and cross-stage coupling. In Tab. 8, both stage and cross-stage coupling methods outperform the uncoupled case. They yield the best results when combined, which is why OLMD incorporates them.

Visualizations

Visualize for LMA. In Fig. 5, we present Grad-CAM (Selvaraju et al. 2016) visualizations of the LMA applied to several sign language videos. The results show that the LMA effectively captures the moving regions (especially the highly

Context	None	3	5	7	9	11
Dev (%)	18.9	18.0	17.7	17.5	17.1	17.3
Test (%)	19.4	18.6	17.8	18.1	17.2	17.5
Params (M)	78.95	80.08	80.30	80.52	80.74	80.97
FLOPs (G)	363.4	407.9	430.3	452.6	474.9	497.2
Time (S)	0.039	0.043	0.045	0.045	0.047	0.048

Table 5: Ablations for context length of LMA on the PHOENIX14. We also record changes in parameters, FLOPs, and inference time (all tested on a 100-frame video).

Operation	Dev (%)	Test (%)
Non-cascaded	17.3	18.1
Cascaded	17.1	17.2

Table 6: Ablations for two designs of OMP on the PHOENIX14.

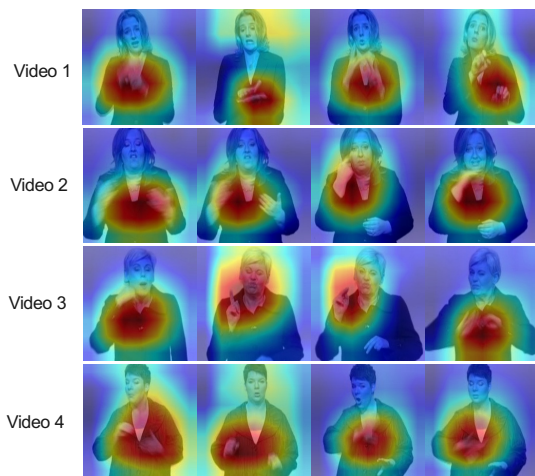


Figure 5: Heatmap visualizations of the LMA module using Grad-CAM (Selvaraju et al. 2016). Obviously, LMA can effectively suppress static information (blue regions) and focus on motion areas (red and yellow regions, indicating moving hands and facial changes).

active hands) while suppressing attention to the background areas. This demonstrates that the LMA is well-designed to capture motion areas and suppress static regions.

Visualize for recognition results. In Fig. 6, we show-case OLMD’s capability to recognize intricate signs. The “Select” involves rapid multi-orientational hand movements, posing a challenge for SEN and CorrNet to recognize it effectively. OLMD addresses this by decoupling motion features into horizontal and vertical components, enhancing orientation awareness through OMP. This approach significantly improves the model’s ability to understand complex motions, enabling accurate recognition of “Select”. The “Why” lasts for 20 frames, which leads SEN and CorrNet to misidentify it as multiple glosses. However, OLMD correctly recognizes “Why” due to the LMA’s capability of aggregating long-term information.

Operation	Dev (%)	Test (%)
Decoupling w/o	18.2	18.5
MaxPool	18.0	18.4
AvgPool	17.1	17.2
MaxPool+AvgPool	17.6	18.0

Table 7: Ablations for decoupling methods on the PHOENIX14.

Operation	Dev (%)	Test (%)
Coupling w/o	18.3	18.5
Stage	17.9	18.3
Cross-stage	17.8	18.2
Stage + Cross-stage	17.1	17.2

Table 8: Ablations for stage and cross-stage coupling on the PHOENIX14.

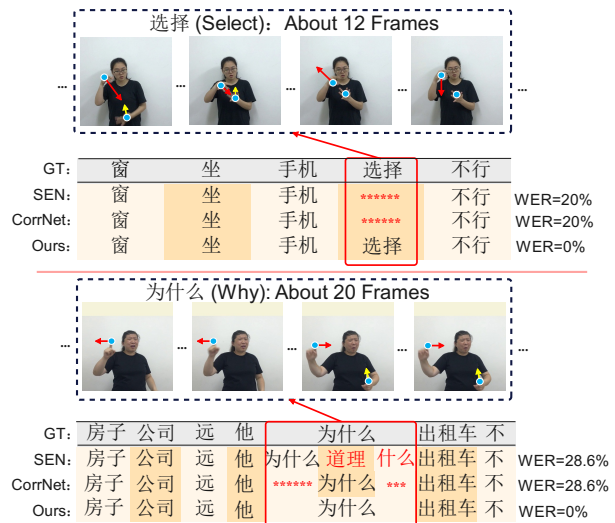


Figure 6: Visualizations of two recognition examples on CSL-Daily (errors in red). **Above:** “Select” involves complex multi-orientational motions. SEN and CorrNet lacked orientation awareness, leading to incorrect recognition. OLMD enhances orientation awareness through decoupling and orientation-aware motion purification, enabling accurate recognition. **Below:** “Why” spans 20 frames. SEN and CorrNet failed to aggregate long-term motion, resulting in multiple glosses and increased errors. The LMA module enables OLMD to recognize this gloss accurately.

Conclusion

This paper presents the OLMD, which markedly advances continuous sign language recognition (CSLR). The proposed LMA effectively aggregates long-term motions, enabling comprehensive sign capture. Furthermore, we enhance the model’s orientation awareness through decoupling and orientation-aware motion purification. Finally, stage and cross-stage coupling finalize the decoupling-coupling process, enhancing the robustness and accuracy. Experimental results on three large datasets demonstrate that OLMD significantly outperforms existing methods, establishing a robust new baseline for future research in CSLR.

Acknowledgement

This work is supported by the Natural Science Foundation of Zhejiang Province (LGG21F030011) and the National Key R&D Program of China (No.2018YFB1305200).

References

- Abdullahi, S. B.; and Chamnongthai, K. 2022. American Sign Language Words Recognition of Skeletal Videos Using Processed Video Driven Multi-Stacked Deep LSTM. *Sensors (Basel, Switzerland)*, 22.
- Bian, G.-B.; Zhang, L.; Chen, H.; Li, Z.; Fu, P.; Yue, W.-Q.; Luo, Y.-W.; Ge, P.-C.; and Liu, W.-P. 2023. Motion Decoupling Network for Intra-Operative Motion Estimation Under Occlusion. *IEEE Transactions on Medical Imaging*, 42(10): 2924–2935.
- Buehler, P.; Zisserman, A.; and Everingham, M. 2009. Learning sign language by watching TV (using weakly aligned subtitles). In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2961–2968.
- Camgoz, N. C.; Hadfield, S.; Koller, O.; Ney, H.; and Bowden, R. 2018. Neural Sign Language Translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7784–7793.
- Chen, Y.; Zuo, R.; Wei, F.; Wu, Y.; Liu, S.; and Mak, B. K.-W. 2022. Two-Stream Network for Sign Language Recognition and Translation. *ArXiv*, abs/2211.01367.
- Cooper, H.; and Bowden, R. 2009. Learning signs from subtitles: A weakly supervised approach to sign language recognition. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2568–2574.
- Cui, R.; Liu, H.; and Zhang, C. 2019. A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training. *IEEE Transactions on Multimedia*, 21: 1880–1891.
- Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.
- Fenlon, J.; Schembri, A.; Johnston, T.; and Cormier, K. 2013. Variation in handshape and orientation in British Sign Language: The case of the '1' hand configuration. *Language & Communication*, 33(1): 69–91.
- Forster, Jens; Ney, Hermann; Koller, and Oscar. 2015. Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. *Computer vision and image understanding: CVIU*, 141: 108–125.
- Gan, S.; Yin, Y.; Jiang, Z.; Wen, H.; Xie, L.; and Lu, S. 2024. SignGraph: A Sign Sequence is Worth Graphs of Nodes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.
- Graves, A.; Fernández, S.; Gomez, F.; and Schmidhuber, J. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, 369–376.
- Guo, L.; Xue, W.; Guo, Q.; Liu, B.; Zhang, K.; Yuan, T.; and Chen, S. 2023. Distilling Cross-Temporal Contexts for Continuous Sign Language Recognition. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10771–10780.
- Hao, A.; Min, Y.; and Chen, X. 2021. Self-Mutual Distillation Learning for Continuous Sign Language Recognition. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 11283–11292.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.
- He, X.; Huang, Q.; Zhang, Z.; Lin, Z.; Wu, Z.; Yang, S.; Li, M.; Chen, Z.; Xu, S.; and Wu, X. 2024. Co-Speech Gesture Video Generation via Motion-Decoupled Diffusion Model. *ArXiv*, abs/2404.01862.
- Hu, L.; Gao, L.; Liu, Z.; and Feng, W. 2022a. Self-Emphasizing Network for Continuous Sign Language Recognition. In *AAAI Conference on Artificial Intelligence*.
- Hu, L.; Gao, L.; Liu, Z.; and Feng, W. 2022b. Temporal Lift Pooling for Continuous Sign Language Recognition. In *European Conference on Computer Vision*.
- Hu, L.; Gao, L.; Liu, Z.; and Feng, W. 2023a. Continuous Sign Language Recognition with Correlation Network. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2529–2539.
- Hu, L.; Gao, L.; Liu, Z.; Pun, C.-M.; and Feng, W. 2023b. AdaBrowse: Adaptive Video Browser for Efficient Continuous Sign Language Recognition. *Proceedings of the 31st ACM International Conference on Multimedia*.
- Huang, J.; gang Zhou, W.; Zhang, Q.; Li, H.; and Li, W. 2018. Video-based Sign Language Recognition without Temporal Segmentation. In *AAAI Conference on Artificial Intelligence*.
- Jiao, P.; Min, Y.; Li, Y.; Wang, X.; Lei, L.; and Chen, X. 2023. CoSign: Exploring Co-occurrence Signals in Skeleton-based Continuous Sign Language Recognition. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 20619–20629.
- Koller, O.; Camgoz, N. C.; Ney, H.; and Bowden, R. 2020. 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: 2306–2320.
- Koller, O.; Zargaran, S.; and Ney, H. 2017. Re-Sign: Re-Aligned End-to-End Sequence Modelling with Deep Recurrent CNN-HMMs. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3416–3424.
- Koller, O.; Zargaran, S.; Ney, H.; and Bowden, R. 2016. Deep Sign: Hybrid CNN-HMM for Continuous Sign Language Recognition. In *British Machine Vision Conference*.
- Lu, H.; Salah, A. A.; and Poppe, R. 2024. TCNet: Continuous Sign Language Recognition from Trajectories and Correlated Regions.

- Min, Y.; Hao, A.; Chai, X.; and Chen, X. 2021. Visual Alignment Constraint for Continuous Sign Language Recognition. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 11522–11531.
- Niu, Z.; and Mak, B. K.-W. 2020. Stochastic Fine-Grained Labeling of Multi-state Sign Glosses for Continuous Sign Language Recognition. In *European Conference on Computer Vision*.
- Qiu, Z.; Yao, T.; Ngo, C.-W.; Tian, X.; and Mei, T. 2019. Learning Spatio-Temporal Representation With Local and Global Diffusion. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12048–12057.
- Reagan, T. 2007. SIGN LANGUAGE AND LINGUISTIC UNIVERSALS, Wendy Sandler and Diane Lillo-Martin. *Studies in Second Language Acquisition*, 29(4): 624–625.
- Selvaraju, R. R.; Das, A.; Vedantam, R.; Cogswell, M.; Parikh, D.; and Batra, D. 2016. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *International Journal of Computer Vision*, 128: 336 – 359.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention Is All You Need. *arXiv*.
- Wang, F.; Guo, D.; Li, K.; Zhong, Z.; and Wang, M. 2024. Frequency decoupling for motion magnification via multi-level isomorphic architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18984–18994.
- Yin, F.; Chai, X.; and Chen, X. 2016. Iterative Reference Driven Metric Learning for Signer Independent Isolated Sign Language Recognition. In *European Conference on Computer Vision*.
- Zheng, J.; Wang, Y.; Tan, C.; Li, S.; Wang, G.; Xia, J.; Chen, Y.; and Li, S. Z. 2023. CVT-SLR: Contrastive Visual-Textual Transformation for Sign Language Recognition with Variational Alignment. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 23141–23150.
- Zhou, H.; Zhou, W.; Qi, W.; Pu, J.; and Li, H. 2021. Improving Sign Language Translation with Monolingual Data by Sign Back-Translation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1316–1325.
- Zhou, H.; Zhou, W.; Zhou, Y.; and Li, H. 2022. Spatial-Temporal Multi-Cue Network for Sign Language Recognition and Translation. *IEEE Transactions on Multimedia*, 24: 768–779.
- Zuo, R.; and Mak, B. 2022. C2SLR: Consistency-enhanced Continuous Sign Language Recognition. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5121–5130.