Sentence-level Segmentation for Long Sign Language Videos with Captions

Bowen Guo
State Key Laboratory for Novel Software Technology, Nanjing University
China
bowen@smail.nju.edu.cn

Shiwei Gan
State Key Laboratory for Novel Software Technology, Nanjing University
China
sw@smail.nju.edu.cn

Yafeng Yin*
State Key Laboratory for Novel Software Technology, Nanjing University
China
yafeng@nju.edu.cn

Xiao Liu
State Key Laboratory for Novel Software Technology, Nanjing University
China
liuxiaox@smail.nju.edu.cn

Zhiwei Jiang
State Key Laboratory for Novel Software Technology, Nanjing University
China
jzw@nju.edu.cn

Shunmei Meng
Department of Computer Science and Engineering, Nanjing University of Science and Technology
China
mengshunmei@njust.edu.cn

Abstract

In existing Sign Language (SL) research, most datasets and backbone models focus on sentence-level samples. However, the annotated sentence-level SL datasets are rather limited, and it is in great need to expand sentence-level SL datasets. When considering the large-scale long SL videos with captions, we propose a new task, *i.e.*, Sentence-level Sign Language Segmentation (SSLS), which splits the long videos into consecutive sentence-level videos. SSLS is an important and meaningful task, which can greatly reduce the labor costs in data annotation for sentence-level SL datasets. However, SSLS is a very challenging task, since it is rather difficult to accurately find the boundary of each sentence in a long video. To address this issue, we formalize, learn, and optimize the boundaries of sentences step by step. First, to distinguish the boundary and the inside of a sentence, we formalize SSLS as a frame-level classification task and design a boundary annotation scheme. Second, to learn the boundary of each sentence from the long video, we design a multimodal framework, SignBD, which correlates the local features and global features through dual dilated attention, while aligning visual and textual (*i.e.*, sentences) modalities through gated cross-attention. Third, to alleviate the widely existed over-segmentation and under-segmentation problems in segmentation tasks, we propose a boundary optimization strategy, which utilizes the number of sentences provided by captions to optimize (*i.e.*, insert or delete) boundaries based on information uncertainty. Extensive experimental results demonstrate the superiority of our solution. Codes are publicly available at: https://github.com/newbg/Sign-Language-Segmentation.

*Corresponding author.

MM '25, Dublin, Ireland

ACM ISBN 979-8-4007-2035-2/2025/10
https://doi.org/10.1145/3746027.3755080

CCS Concepts

• **Computing methodologies** → **Video segmentation**.

Keywords

Sign Language, Video Segmentation, Video and Text Alignment

ACM Reference Format:
Bowen Guo, Shiwei Gan, Yafeng Yin, Xiao Liu, Zhiwei Jiang, and Shunmei Meng. 2025. Sentence-level Segmentation for Long Sign Language Videos with Captions. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25), October 27–31, 2025, Dublin, Ireland.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3746027.3755080

1 Introduction

Current research in Sign Language (SL) understanding primarily focuses on Continuous Sign Language Recognition (CSLR) [11, 48, 55] and Sign Language Translation (SLT) [12, 42, 51, 53]. The existing CSLR and SLT models [42, 50, 52, 57] heavily rely on publicly available SL datasets, where samples are typically organized in sentence level, *i.e.*, video-gloss pairs (for CSLR) and video-sentence pairs (for SLT). In fact, the existing SL models can hardly work for multi-sentence video processing (see Table 7). That is to say, sentence-level video-text pairs are essential for SL models. However, the scarcity of annotated sentence-level SL datasets severely limits the development of SL tasks. Consequently, it is crucial to explore effective methods for expanding sentence-level SL datasets.

A straightforward method for producing sentence-level datasets is to recruit signers to record the video for each pre-defined sentence. However, the labor cost of this method is very high, and the size of collected dataset is often very small. To reduce the burden of recoding videos, another method is to invite domain experts to manually annotate the existing SL videos in sentence level. The labor cost of annotating videos is also high, and the dataset size is also limited. That is to say, these existing methods are often labor-intensive and tend to produce small-size datasets. To overcome these limitations, we take the lead in exploring the large-scale long sign language videos with captions (*i.e.*, translated text) and automatic data annotation methods. Specifically, we aim to split the

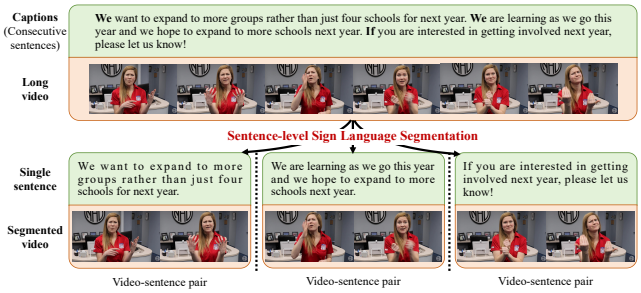


Figure 1: Illustration of sentence-level sign language segmentation.

long videos into sentence-level short videos based on the written sentences in captions, and then construct the video-sentence pairs for sentence-level SL datasets. Due to the large number of long videos, it is possible to greatly expand the scale of sentence-level SL datasets, while significantly reducing labor costs.

To achieve the above goal, we propose a new task, *i.e.*, Sentence-level Sign Language Segmentation (SSLS), as illustrated in Figure 1. In the SSLS task, the objective is to automatically segment a long sign language video to consecutive short videos, where the semantic information of each short video corresponds to a subtitle sentence. Apparently, after SSLS, each segmented video and its corresponding written sentence will form a video-sentence pair, thus we are possible to automatically produce a large-scale sentence-level SL dataset, while greatly reducing labor costs. The larger datasets are beneficial to build large-scale sign language models and further promote the SL research. However, as a new task, to achieve SSLS, there are several challenges to be addressed. First, there are no suitable datasets for SSLS, since the existing datasets are usually annotated for SLR or SLT tasks. Second, it is difficult to find the boundary of short video corresponding to a sentence, since there are usually not apparent transition actions in sign language. Third, it is rather challenging to get the correct number of segments (*i.e.*, short videos), since the over-segmentation and under-segmentation problems [4, 19, 33, 45] are widely existed in segmentation tasks.

In this paper, we seek to solve the above challenges, and then provide an effective baseline model and produce suitable datasets for SSLS tasks. When considering the challenges, we formalize, learn, and optimize the boundaries of short videos step by step, based on the written sentences in captions. First, to define the new task lacking of suitable datasets, we formalize SSLS as a frame-level classification task, and then design a variable-length boundary annotation scheme to distinguish the boundary frames and inside frames of a sentence. Second, to learn the boundary of each short video corresponding to a written sentence for segmentation, we design a multimodal framework, where dual dilated attention is adopted to correlate local features and global features (*i.e.*, locate a local boundary from a global perspective), while gated cross attention is adopted to align visual and textual modalities (*i.e.*, locate boundaries by distinguish different sentences). Third, to alleviate the widely existed over-segmentation and under-segmentation problems in segmentation tasks, we propose a post-processing boundary optimization strategy, which utilizes the number of sentences provided by captions to further insert or delete boundaries, according to the calculated information uncertainty. In this way, we split a long sign language video into consecutive short videos in sentence level, and produce more sentence-level datasets for SL research.

We make the following contributions in this paper.

- We propose a new task, Sentence-level Sign Language Segmentation (SSLS), which plays an important role in expanding sentence-level SL datasets and promoting the SL research. To define the new task and provide suitable training data, we formalize SSLS as a frame-level classification task and design a variable-length boundary annotation scheme.
- To learn the boundary of each short video(*i.e.*, in sentence level) from the long video, we design a multimodal framework, which locates boundaries from a global perspective through dual dilated attention and detects boundaries between different sentences by aligning visual and textual modalities through gated cross-attention.
- To alleviate the over-segmentation and under-segmentation problems, we propose a post-processing boundary optimization strategy, which utilizes the number of written sentences to further insert or delete boundaries based on information uncertainty.
- Extensive experimental results demonstrate the superiority of the proposed model, which can serve as effective method for SSLS. Besides, the segmented video-sentence pairs can be used to expand SL datasets.

2 Related Work

Until now, there has been no research work on sentence-level sign language segmentation (SSLS). Therefore, we will review the work most relevant to SSLS, *i.e.*, , action segmentation, gloss segmentation/alignment, and caption alignment.

2.1 Action Segmentation

Action segmentation [19, 26, 27, 32, 34, 49] aims to partition a long video into a series of short video segments, where each segment corresponds to an action. Usually, action segmentation is formalized as a frame-wise classification task, *i.e.*, each frame is labeled with an action class. The existing methods can be broadly classified into fully supervised multi-stage refinement approaches and weakly supervised alignment-based approaches. The former ones have an action label for each frame in model training. Then, they predict and refine action segments in multiple stages iteratively. For example, MS-TCN [9] and MS-TCN++ [30] designed multiple stages, which are consisted of temporal convolutional layers, to refine action segmentation. ASFormer [54] introduced multiple decoders to refine the predicted action segments in stages. While the latter ones do not have an action label for each frame, and they only have transcripts that list the action order. Thus they usually generated pseudo frame-wise action labels, and aligned video segments with transcripts using algorithms such as Viterbi [24, 25, 28, 35, 36] or Dynamic Time Warping [10, 18, 38, 43, 44]. However, both types of approaches often suffer from over-segmentation (*i.e.*, an action is split into multiple video segments) and under-segmentation issues (*i.e.*, multiple actions are split into a single video segment). To mitigate these issues, ASRF [19] introduced an auxiliary action boundary network to refine segmentation, while some two-stage methods [1, 3, 18, 20, 29] refine predictions by modeling action relations. Different from action segmentation, our SSLS task belongs to

a binary classification problem, *i.e.*, distinguishing intra-clip frames (internal frames) and inter-clip frames (boundary frames). However, our SSLS also suffers from the imbalance between the number of internal frames and boundary frames. Therefore, we propose a multi-frame boundary annotation scheme to alleviate the imbalance problem, and also introduce boundary learning and optimization strategies to improve the segmentation accuracy.

2.2 Gloss-level Segmentation and Alignment

Gloss is the minimal linguistic unit in sign language, and it often corresponds to a word in spoken language [22]. In terms of gloss-level video segmentation, some work introduced Hidden Markov Model (HMM) [13, 15, 16, 22] to infer the temporal segmentation of a clip corresponding to a gloss. For example, DTW-HMM [56] combined HMM-based modeling with a two-stage segmentation strategy, which first uses a threshold matrix to detect coarse sign boundaries, and then applies Dynamic Time Warping (DTW) for fine-grained segmentation through candidate matching and refinement. In terms of gloss-level video-text alignment, some research work tried to align each gloss with the key frames of the corresponding clip [2, 46]. For example, Momeni et al. [39] used dictionary lookups to identify potential gloss occurrences, enabling weak alignment between isolated signs and video clips for sign spotting. Gul Varol et al. [46] used the maximum attention score corresponding to each gloss to temporally locate the key frames. While the other research work often adopted Connectionist Temporal Classification (CTC) loss [14, 40, 58] to weakly align the gloss sequences and video clips in order. In SSLS, the boundary frame of each video clip should be explicitly detected. Thus the existing gloss-level approaches with coarse or weak alignment are not suitable for our task. Therefore, we explore a new framework to formalize, learn and optimize the boundary of each video clip for SSLS.

2.3 Sentence-level Caption Alignment

Sentence-level caption alignment aims to align written sentences with video segments, when given the text captions of a video. Some work attempted to detect boundaries by relying on visual cues such as hand movements, pauses, and facial expressions [21, 41]. Bull et al. [6] used visual skeleton data, including body, hand, and face keypoints to detect temporal sign boundaries, and aligned them with provided subtitles. Subsequent work considered the practical scenario in which subtitles are available, for instance, Bull et al. [5] used caption timestamps (may be not accurate) of written sentences as priors to define temporal windows and incorporated textual cues for alignment. In contrast, our SSLS framework directly operates on long sign language videos without relying on subtitle timestamps, and even provides visual-only working modes without captions.

3 Problem Definition

Sentence-Level Sign Language Segmentation (SSLS) is a new task and can be defined as follows. Suppose there is a long, untrimmed sign language video $F = \{f_i\}_{i=1}^{\theta}$, where f_i denotes the i-th video frame and θ is the total number of frames. The corresponding text caption of the long video is $S = \{s_j\}_{j=1}^{\varsigma}$, where s_j is the j-th sentence and ς denotes the total number of written sentences. Then, the goal of the SSLS task is to segment F into a set of non-overlapping sub-videos $V = \{v_j\}_{j=1}^{\varsigma}$, where $v_j = \{f_{a_j}, f_{a_j+1}, \ldots, f_{b_j}\}$ is semantically aligned with the sentence s_j and satisfies $1 \le a_j \le b_j \le \theta$. It is worth noting that due to the widely existed over-segmentation and under-segmentation problems, the number of segmented clips ε may not be equal to ς, and the jth sub-video v_j my not be aligned with the jth sentence s_j.

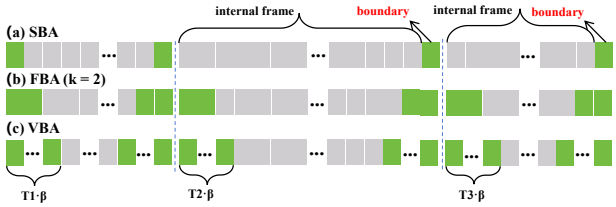


Figure 2: Comparison of three annotation strategies: (a) SBA; (b) FBA with $k = 2$; (c) VBA.

To segment the long video into short clips, the key is to detect the boundaries (*i.e.*, start and end) of each clip. To detect the boundary from all video frames, we re-formalize the SSLS task as a frame-wise binary classification problem, *i.e.*, classifying a frame as an internal frame or a boundary frame. Specifically, the objective of SSLS is to predict a frame-wise class sequence $Z = \{z_i | z_i \in \{0, 1\}\}_{i=1}^{\theta}$, where $z_i = 0$ denotes an internal frame and $z_i = 1$ indicates a boundary frame. Suppose that $X = \{x_i\}_{i=1}^{\theta}$ corresponds to the frame-level visual feature sequence, then the neural model needs to learn the conditional probability $\mathrm{P}(Z \mid X)$. When the class sequence Z is learned, the k-th video segment v_k is represented with consecutive frames $(f_{k_1}, f_{k_p}]$, where f_{k_1} and f_{k_p} are boundary frames (*i.e.*, $z_{k_1} = 1, z_{k_p} = 1$), and $f_{k_i}, i \in (1, p)$ are internal frames (*i.e.*, $z_{k_i} = 0$). After that, the k-th video segment and the k-th written sentence will form a video-sentence pair, as shown in Figure 1.

4 Method

To solve the challenging SSLS task, we define, learn, and optimize the boundaries of sub-videos step by step. First, we design a multi-frame boundary annotation scheme to distinguish internal frames and boundary frames of a video segment corresponding to a written sentence, as illustrated in Figure 2. Second, we propose a neural Sign language segmentation framework based on Boundary Detection (named SignBD for short), to learn the boundary of each video segment as shown in Figure 3. Third, we present a segmentation optimization module, to optimize the predicted video segments, by calculating the information uncertainty of each frame and comparing the number of segments and that of sentences in text captions.

4.1 Boundary Definition

In SSLS, a long sign language video often contains thousands of frames but only a few sentence-level segments, leading to a severe imbalance in the number of internal and boundary frames. This imbalance causes the model to be dominated by internal frames during training, making it harder to recognize boundary patterns. To alleviate this issue, we propose a boundary annotation strategy named Variable-length Boundary Annotation (VBA), which expands the boundary regions around the transition of each segment.

Figure 2 illustrates the difference between the conventional Single-frame Boundary Annotation (SBA), Fixed-frame Boundary Annotation (FBA), and our proposed VBA. In this example, the

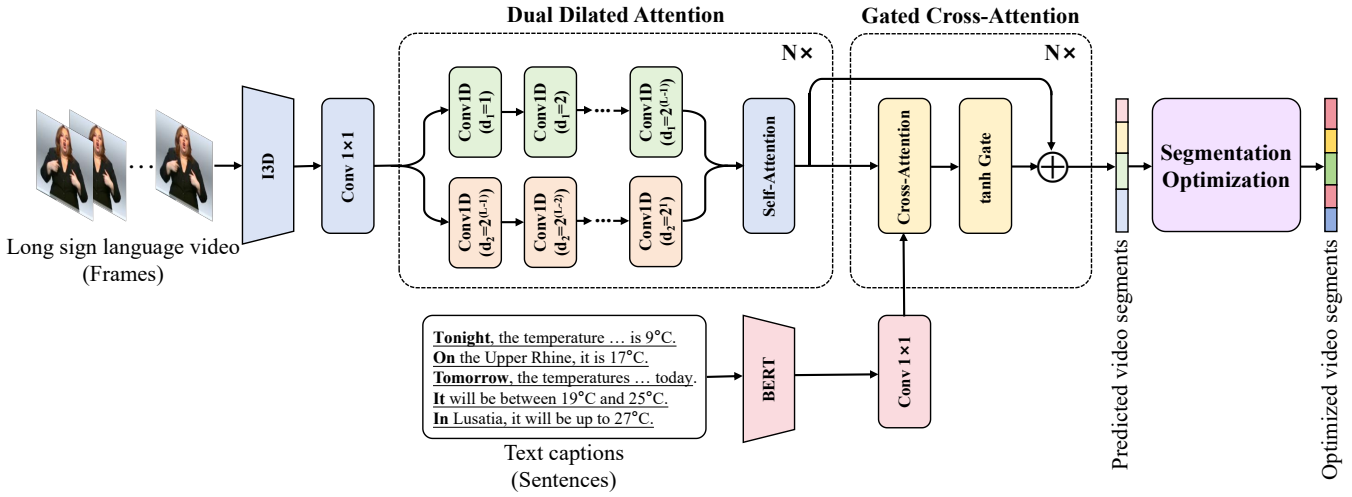


Figure 3: The details of proposed SignBD framework.

video is segmented into three parts with lengths T_1, T_2, and T_3. As shown in Figure 2(a), SBA labels only the last frame of each segment, along with the first frame of the video, as boundary frames ($z = 1$). FBA, illustrated in Figure 2(b), labels fixed k frames at both sides of each segment, regardless of segment length. In contrast, VBA, shown in Figure 2(c), adaptively labels $\lceil T_i \cdot \beta \rceil$ frames on both the start and end sides of each segment v_i. The remaining frames are labeled as internal ($z = 0$). VBA adapts to varying segment lengths, making it more suitable than SBA and FBA.

4.2 Boundary Learning

In long sign language videos, transitions between consecutive sentences are often subtle and lack apparent visual boundaries. This makes it difficult to accurately locate segment transitions based solely on visual features. To tackle this challenge, we introduce a boundary learning framework named SignBD that incorporates sentence-level captions as a auxiliary modality to guide the learning of sentence boundaries.

As shown in Figure 3, our framework takes a long sign language video and its corresponding captions as input. The video is first encoded by an I3D model to obtain frame-wise visual features. A 1×1 convolution is then applied for dimension reduction. The features are passed into a Dual Dilated Convolution (DDC) module, which consists of two parallel convolutional paths with increasing and decreasing dilation factors. The output of DDC is then passed to 10 stacked self-attention layers, which model global dependencies across the entire sequence. Meanwhile, the caption is encoded using a pre-trained BERT model and projected via a 1×1 convolution to match the dimension of visual features. The caption features are integrated with the visual features through a Gated Cross-Attention module, which aligns the visual and textual representations, ensuring that the sentence-level semantics guide the boundary learning process. The resulting sequence of frame-wise boundary predictions is further processed by the segmentation optimization (SegOpt) module, which leverages frame-wise information uncertainty values and the number of sentences to correct over-segmentation and under-segmentation errors. The final output is a frame-wise binary sequence, which indicates whether a frame is a boundary frame or an internal frame, thus getting the segmented video clips.

Dual Dilated Attention. The Dual Dilated Attention (DDA) module consists of the Dual Dilated Convolution (DDC) module and stacked self-attention layers. Given a visual feature sequence $X \in \mathbb{R}^{C \times L}$, where C is the feature dimension and L is the temporal length, the DDC employs two parallel convolutional paths with complementary dilation factors for encoding. In the first path, the dilation factors d_1 grow exponentially with layer depth (e.g., $1, 2, 4, 8, 16, \ldots$), enabling the capture of long-range dependencies. In contrast, the dilation factors in the second path d_2 decay exponentially (e.g., $\ldots, 16, 8, 4, 2, 1$), emphasizing local details. The outputs of the two paths are computed as:

$$X_1 = \text{Conv1D}(X; d_1), \quad X_2 = \text{Conv1D}(X; d_2), \tag{1}$$

where $X_1, X_2 \in \mathbb{R}^{C \times L}$ are the feature maps generated by two convolutional paths, respectively. These outputs are first concatenated along the channel dimension and then passed through a 1D convolutional layer to reduce the dimensionality:

$$X_{\text{dilated}} = X_1 \oplus X_2, \tag{2}$$

where $X_{\text{dilated}} \in \mathbb{R}^{C \times L}$ captures both short-range and long-range temporal patterns. The fused feature is then passed through the self-attention module to model global dependencies:

$$X_{\text{DDA}} = \text{Self-Attention}(X_{\text{dilated}}), \tag{3}$$

where $X_{\text{DDA}} \in \mathbb{R}^{C \times L}$ denotes the final output of the DDA module. The DDA module allows the model to extract both local temporal patterns and long-range dependencies from different receptive fields, thereby capturing transitions between consecutive sentences.

Gated Cross-Attention. To incorporate sentence-level semantic guidance, we align the visual features with caption features using a Gated Cross-Attention mechanism. The caption features $T \in \mathbb{R}^{C_t \times S}$, with C_t being the textual feature dimension and S being the number of tokens, are projected to match the visual dimension. Then, cross-attention is performed with T as queries and X_{DDA} as

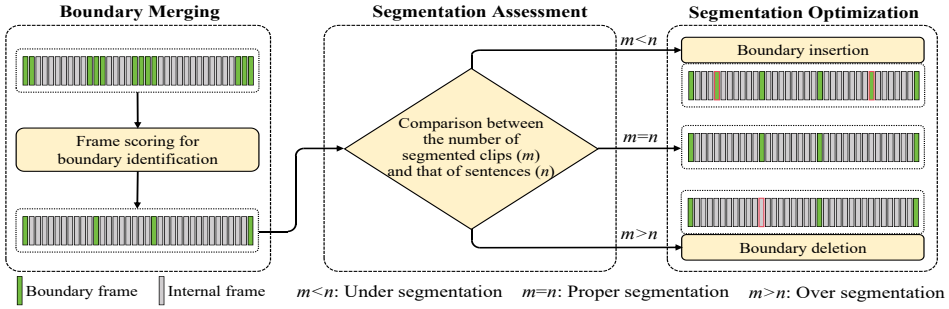


Figure 4: Illustration of the proposed SegOpt module.

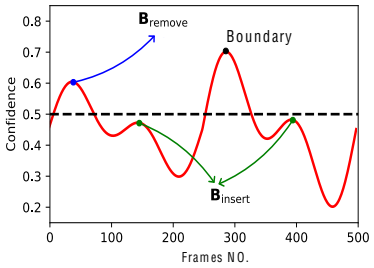


Figure 5: Frame-wise confidence.

keys and values. The output is combined with the original visual features through a gating mechanism:

$$X_{\text{fused}} = \tanh(\alpha) \cdot \text{CrossAttn}(T, X_{\text{DDA}}) + (1 - \tanh(\alpha)) \cdot X_{\text{DDA}}, \quad (4)$$

where α is a learnable scalar parameter, and $\tanh(\alpha)$ is the hyperbolic tangent function, which serves as a soft gate to balance the contributions of the text-guided visual features and the original visual representations. After that, the fused features are used to predict frame-wise boundary probabilities. This mechanism aligns semantic and visual information at each frame, enabling more accurate boundary prediction. The predictions are further processed by the SegOpt module described in Section 4.3.

Loss Function. Following prior works on action segmentation [9, 30, 37, 54], we adopt a loss function that combines Cross-Entropy loss and Smooth Loss to supervise frame-wise boundary prediction. Specifically, the model outputs a probability matrix $\hat{Y} \in \mathbb{R}^{2\times L}$, where $\hat{y}_{c,i}$ in $\hat{Y}$ denotes the predicted probability of class $c \in \{0, 1\}$ at frame i, and L is the total number of frames. The ground truth labels are given by $Z = \{z_1, z_2, \ldots, z_L\}$, where $z_i \in \{0, 1\}$. Then, the Cross-Entropy Loss is defined with Eq. (5), while the Smooth Loss is defined with Eq. (6). After that, we combine the losses of both encoder and decoder, and get the total loss with Eq. (7), where $\lambda = 0.15$ is used to control the smoothness weight.

$$\mathcal{L}_{\text{CE}} = -\frac{1}{L} \sum_{i=1}^{L} \log \hat{y}_{z_i,i}, \quad (5)$$

$$\mathcal{L}_{\text{smooth}} = \frac{1}{L-1} \sum_{i=1}^{L-1} \sum_{c=1}^{2} \left\| \hat{y}_{c,i+1} - \hat{y}_{c,i} \right\|_2^2. \quad (6)$$

$$\mathcal{L} = \mathcal{L}_{\text{CE}}^{(enc)} + \lambda \mathcal{L}_{\text{smooth}}^{(enc)} + \mathcal{L}_{\text{CE}}^{(dec)} + \lambda \mathcal{L}_{\text{smooth}}^{(dec)}, \quad (7)$$

4.3 Boundary Optimization

Most segmentation tasks inevitably suffer from over-segmentation and under-segmentation issues. Suppose we get m video segments after segmentation, while there are n sentences in the text caption. If $m < n$, then we suffer from under-segmentation issues. If $m > n$, then we suffer from over-segmentation issue. To mitigate these issues in SSLS, we propose a segmentation optimization module named SegOpt, as illustrated in Figure 4. This module first evaluates the reliability of predicted boundaries based on information uncertainty to construct a candidate boundary set, and then utilizes the number of sentences from the caption to guide the insertion and deletion of boundaries. The module consists of three steps: frame scoring, boundary merging, segmentation assessment and optimization.

Frame Scoring. The proposed SignBD model produces frame-wise confidence scores $C = [C_1, C_2, \ldots, C_T]$, where C_t denotes the likelihood of frame f_t being a boundary. To suppress noise and enhance temporal stability, we apply a Gaussian filter to smooth the confidence curve (shown in Figure 5). This reduces local fluctuations while preserving prominent boundary signals across the entire video. We further calculate information uncertainty using the gradient and local extrema of the smoothed confidence. The uncertainty U_t for the tth frame is computed as:

$$U_t = \left| \frac{\partial C_t}{\partial t} \right| + \epsilon \cdot \max\left(C_t - \frac{C_{t-1} + C_{t+1}}{2}, 0 \right), \quad (8)$$

where the first term captures the temporal variation of the confidence value, and the second term (weighted by the hyperparameter ϵ) emphasizes local confidence peaks by measuring their deviation from the local average. Sentence boundaries in sign language videos often correspond to sharp changes and local maxima in the frame-wise confidence scores. In particular, consecutive high-confidence values tend to produce temporally continuous boundary predictions, and prominent peaks in the confidence curve serve as key indicators of potential boundaries. Therefore, a larger U_t, which captures both the gradient and local peak of the confidence curve, indicates a higher likelihood that frame t is a boundary, as illustrated in Figure 5.

Boundary Merging. During inference, our VBA strategy often leads to multi-frame boundary predictions, where boundary frames appear consecutively. To facilitate subsequent optimization and evaluation, these predictions are converted into single-frame boundaries, by selecting the frame having the highest uncertainty value within each continuous boundary region as the final predicted boundary.

Segmentation Assessment and Optimization. Based on the difference between the predicted number of segments m and the expected number n, we propose a segmentation optimization strategy that dynamically adjusts the number and position of predicted boundaries. For under-segmentation ($m < n$), we insert additional boundaries by identifying frames with low confidence and high information uncertainty. First, based on the confidence, the candidate set is defined as:

$$\mathcal{B}_{\text{insert}} = \{ t \mid C_t \in (\max(C_{t-1}, C_{t+1}),\ \tau) \}, \quad (9)$$

Dataset	Train		Dev		Test	
	Num	Avg sentences	Num	Avg sentences	Num	Avg sentences
PHOENIX-2014T	822	8.07	102	8.27	104	7.52
How2Sign	3720	6.52	461	5.71	464	5.48
OpenASL	11023	6.68	1378	6.69	1379	6.79

Table 1: Statistics of the sign language datasets.

where $k = |\mathcal{B}_{\text{insert}}|$ is the number of candidate frames for insertion, and t denotes the index of the t-th frame in the video sequence. Then, the final boundaries to insert are the $\min(k, n-m)$ frames with the highest uncertainty values:

$$\mathcal{B}_{\text{insert}}^{\text{final}} = \underset{t \in \mathcal{B}_{\text{insert}}}{\text{argmax}}^{\min(k,n-m)} U_t. \tag{10}$$

For over-segmentation ($m > n$), we remove redundant boundaries by evaluating predicted boundaries with high confidence and low uncertainty. The candidate set is:

$$\mathcal{B}_{\text{remove}} = \{t \mid C_t \in (\max\{\tau, C_{t-1}, C_{t+1}\},\ 1)\} \tag{11}$$

The final boundaries to remove are the $(m-n)$ frames with the lowest uncertainty values:

$$\mathcal{B}_{\text{remove}}^{\text{final}} = \underset{t \in \mathcal{B}_{\text{remove}}}{\text{argmin}}^{(m-n)} U_t. \tag{12}$$

This module adjusts the predicted number of segments to match the expected number of sentences, thereby alleviating over-segmentation and under-segmentation issues in SSLS. In our implementation, the threshold τ is set to 0.5.

5 Experiments and Results

5.1 Datasets

To adapt to the proposed SSLS task, we processed three publicly available sign language datasets: PHOENIX-2014T[7], How2Sign[8], and OpenASL[8], which can be categorized into two types: manually synthesized videos and naturally continuous videos. The detailed statistics are provided in Table 1. The PHOENIX-2014T [7] contains sentence-level sign language videos recorded for German weather forecasts. To adapt it to SSLS, we grouped and concatenated individual sentence videos by date, resulting in 822 training, 102 validation, and 104 test samples, each containing approximately 8 sentences on average. The other two datasets, How2Sign [8] and OpenASL [8], consist of naturally continuous sign language videos. We directly split raw videos into 1–2 minute videos containing multiple sentences. From How2Sign, we obtained 3720 training, 461 validation, and 464 test samples. From OpenASL, we obtained 11023 training, 1378 validation, and 1379 test samples. Each video contains around 5–7 sentences on average. Using the proposed VBA annotation strategy, we processed these datasets to create benchmarks specifically suited for the SSLS task.

5.2 Experimental Setup

Data Preprocessing. Follow the existing work [17, 38], we applied several data augmentation techniques, which include resizing frames to 256×256 pixels, randomly cropping them to 224×224 pixels, applying horizontal flipping with a probability of 0.5, and random temporal scaling with a factor sampled from the range $[0.8, 1.2]$.

Model	Dev				Test			
	F1@10	F1@25	F1@50	SER	F1@10	F1@25	F1@50	SER
MS-TCN [9]	89.07	88.61	80.66	0.133	89.65	89.25	81.18	0.086
MS-TCN++ [30]	89.01	88.34	79.83	0.091	90.77	90.37	81.82	0.086
ASformer + MLP [54]	82.95	80.00	55.92	0.074	82.67	80.37	56.90	0.087
ASFormer + Conv [54]	88.62	87.73	79.67	0.132	89.18	88.45	80.96	0.110
FACT [37]	90.74	89.36	80.78	0.131	89.73	88.84	81.16	0.169
SignBD (ours)	**92.99**	**91.79**	**84.55**	**0.036**	**93.94**	**92.65**	**87.58**	**0.036**

Table 2: Evaluation results on the PHOENIX-2014T dataset.

Architecture Settings. Our framework is implemented using PyTorch 1.13. We adopt the proposed VBA strategy with an annotation proportion $\beta = 0.2$. Visual features are extracted using a pre-trained I3D model [47], yielding 1024-dimensional frame-wise features. Textual features are obtained from a pre-trained BERT model [23], with a feature dimension of 768. The visual stream is processed by a 12-layer DDA module, followed by a 12-layer Gated Cross-Attention module for modality alignment. For SegOpt, a Gaussian filter with $\sigma = 5$ is used to smooth the confidence curve, and a 1D convolutional layer with kernel size 3 is used to adjust the feature dimensionality.

Training Configuration. We train our model for 60 epochs using the Adam optimizer with a weight decay of 0.0001. The initial learning rate is set to 0.0005, and the batch size is 6. All experiments are conducted on 4 GeForce RTX 4090 GPUs.

Evaluation Metrics. To evaluate the performance of our proposed SignBD, we utilize two primary metrics:

Segmental F1 Score: Following prior works in action segmentation [9, 54], we compute the segment-level F1 score at overlapping threshold of 10%, 25%, and 50%, denoted as F1@10, F1@25, and F1@50, respectively.

Segment Error Rate (SER): We propose a new metric, SER, to measure the difference between the predicted number of segments $\varepsilon_{\text{pred}}$ and the actual number of sentences ε_{gt}. It is computed as:

$$\text{SER} = \frac{|\varepsilon_{\text{pred}} - \varepsilon_{\text{gt}}|}{\varepsilon_{\text{gt}}}, \tag{13}$$

A lower SER indicates better alignment between the number of predicted and ground truth sentence boundaries.

Baselines. We reproduce several representative models for action segmentation, including MS-TCN [9], MS-TCN++ [30], ASFormer [54], DiffAct [31], and FACT [37]. For ASFormer, we consider both variants that differ in the design of the feed-forward module: one using convolutions and the other using MLPs. We modify the original segmentation heads, designed to assign frame-level labels from a predefined action set, into binary classifiers for sentence boundary detection.

5.3 Overall Performance

We evaluate the proposed SignBD on the validation ('DEV') and test ('TEST') sets of three benchmark datasets.

Evaluation on PHOENIX-2014T. As shown in Table 2, SignBD achieves the best performance across all F1 metrics on both the DEV and TEST sets. In particular, SignBD outperforms the SOTA

Model	How2Sign								OpenASL							
	Dev				Test				Dev				Test			
	F1@10	F1@25	F1@50	SER	F1@10	F1@25	F1@50	SER	F1@10	F1@25	F1@50	SER	F1@10	F1@25	F1@50	SER
MS-TCN [9]	60.61	58.45	38.04	0.558	61.56	59.71	39.94	0.428	76.60	75.25	62.61	0.505	77.77	76.61	64.24	0.208
MS-TCN++ [30]	69.29	67.62	48.03	0.406	69.06	67.37	48.65	0.428	74.65	73.23	60.72	0.257	76.23	74.96	62.22	0.245
ASformer + MLP [54]	51.42	47.01	24.47	0.679	51.56	47.29	26.42	0.683	70.46	67.13	42.93	0.608	69.80	66.49	41.52	0.669
ASFormer + Conv [54]	71.34	69.41	51.43	0.385	71.61	69.70	51.89	0.394	79.31	78.12	65.83	0.206	80.45	79.28	67.74	0.196
FACT [37]	80.26	78.30	56.66	0.340	80.89	78.99	58.43	0.323	84.77	83.12	69.89	0.229	85.03	83.24	70.19	0.217
DiffAct [31]	83.12	80.94	60.87	0.2990	82.76	80.41	60.35	0.294	81.68	79.39	60.40	0.3218	81.25	78.90	59.92	0.317
SignBD (ours)	**88.34**	**85.91**	**68.38**	**0.128**	**89.11**	**86.21**	**69.18**	**0.098**	**88.68**	**86.33**	**73.69**	**0.016**	**89.19**	**86.91**	**74.66**	**0.002**

Table 3: Evaluation results on How2Sign and OpenASL datasets.

	Method	F1@10	F1@25	F1@50	SER
PHOENIX-2014T	w/o SegOpt	88.14	87.91	80.50	0.170
	w/ SegOpt	93.36	91.91	86.49	0.031
How2Sign	w/o SegOpt	72.62	71.02	53.14	0.403
	w/ SegOpt	89.11	86.21	69.18	0.098

Table 4: Effect of the SegOpt on sentence-level segmentation performance.

	Method	F1@10	F1@25	F1@50	SER
PHOENIX-2014T	w/o text	91.85	90.33	84.62	0.121
	w/ text	93.94	92.65	87.58	0.036
How2Sign	w/o text	80.31	78.02	59.85	0.252
	w/ text	89.11	86.21	69.18	0.098

Table 5: Effect of incorporating text features on sentence-level segmentation performance.

baseline FACT by 6.42% in F1@50 on the test set. In terms of the number of segments, SignBD also achieves the lowest SER of 0.036, outperforming all other baselines.

Evaluation on How2Sign. Table 3 (left) shows that SignBD significantly outperforms all baselines on the How2Sign dataset. It achieves the highest F1@50 on both DEV and TEST sets, surpassing the strongest baseline DiffAct by 7.51% and 8.83%, respectively. Further, SignBD also achieves a much lower SER of 0.098 on the test set, compared to 0.294 from DiffAct.

Evaluation on OpenASL. As shown in Table 3 (right), SignBD also outperforms all baseline methods on OpenASL. It achieves a notable F1@50 of 74.66% on the test set, exceeding the FACT by 4.47%. Moreover, SignBD attains the lowest SER of only 0.002, which is significantly better than all other models.

Strategy	F1@10	F1@25	F1@50
SBA	26.19	9.27	0.00
FBA (k=5)	64.22	60.86	48.21
FBA (k=10)	86.85	86.31	77.93
FBA (k=20)	86.10	85.61	76.75
VBA (β=0.2)	**93.94**	**92.65**	**87.58**

Table 6: Comparison of annotation strategies on How2Sign.

5.4 Ablation study

Effect of DDA. To verify the effectiveness of DDA module, we compare it with two variants of the feed-forward layer: a multi-layer perceptron (MLP) and a dilated convolution, on How2Sign. To ensure a fair comparison, all variants are evaluated without applying the SegOpt module. As shown in Figure 6, our DDA outperforms the other two variants across all F1 metrics. Notably, it achieves 53.14% on F1@50, which is significantly higher than the dilated convolution (49.72%) and MLP (26.42%) counterparts. This because that DDA can more effectively capture both local temporal details and long-range dependencies for accurate sentence boundary segmentation.

Effect of SegOpt. To evaluate the effectiveness of SegOpt, we compare models with and without SegOpt on PHOENIX-2014T and How2Sign, as shown in Table 4. On PHOENIX2014T, F1@50 increases from 80.50% to 86.49%, and SER decreases from 0.170 to 0.031. On How2Sign, F1@50 improves from 53.14% to 69.18%, and SER drops from 0.403 to 0.098. These results demonstrate that SegOpt improves segmentation performance by alleviating over-segmentation and under-segmentation.

Effect of Captions. To evaluate the effect of captions, we conduct experiments on PHOENIX-2014T and How2Sign with and without textual guidance, as shown in Table 5. On PHOENIX-2014T, F1@50 increases from 84.62% to 87.58%, and SER drops from 0.011 to 0.005. On How2Sign, the improvement is more pronounced, with F1@50 rising from 59.85% to 69.18% and SER decreasing from 0.252 to 0.098. These improvements can be attributed to the additional semantic information provided by captions, which enables the model to more accurately identify ambiguous boundaries. SignBD effectively leverages captions when available, while still working well in a visual-only mode.

Effect of VBA. We compare different annotation strategies and find that VBA significantly outperforms both SBA and FBA, as shown in Table 6. This is because assigning a fixed number of boundary frames in FBA introduces excessive annotation noises for short segments, while failing to balance the number of boundaries and internal frames for long segments. To select a proper hyperparameter β for VBA, we evaluate performance on PHOENIX-2014T and How2Sign with $\beta \in \{0.025, 0.05, 0.1, 0.2, 0.3\}$, as shown in figure 7. As β increases, the F1 scores initially improve, reaching the

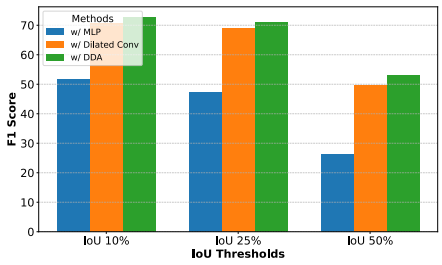
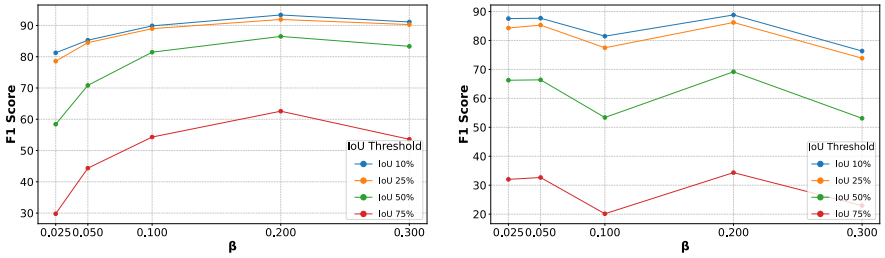


Figure 6: Effect of the DDA Module.



(a) On the PHOENIX-2014T dataset

(b) On the How2Sign dataset

Figure 7: Exploring the effect of the hyperparameter β .

highest performance at $\beta = 0.2$, and subsequently decline. We therefore set $\beta = 0.2$ in all experiments.

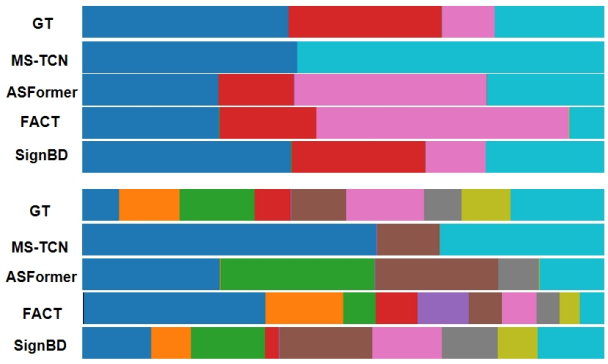


Figure 8: Qualitative comparison of segmentation results on the How2Sign. From top to bottom: Ground Truth (GT), MS-TCN, ASFormer, FACT and SignBD (ours).

Figure 9: Visualization of segmentation results with and without SegOpt module.

Setting	Rouge	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Single-sentence	52.65	53.97	41.75	33.84	28.39
Multi (4, pre-trained)	19.99	20.43	11.35	7.67	5.58
Multi (4, re-trained)	27.24	24.13	14.50	9.36	6.72

Table 7: SLT performance on single- and multi-sentence sign language videos on How2Sign.

5.5 Qualitative Analysis

Figure 8 visualizes the predictions of different models on How2Sign. MS-TCN tends to under-segment, while ASFormer and FACT show large boundary deviations. In contrast, SignBD achieves more accurate segmentation results, because our framework aligns captions with visual features and the SegOpt module that refines predictions. Additionally, Figure 9 visualizes the effect of SegOpt, showing that it can effectively correct both over-segmentation and under-segmentation.

6 Discussion

SLT models have typically focused on sentence-level videos and perform well on shorter inputs. We evaluate SLT performance on both single-sentence and multi-sentence sign language videos, as shown in Table 7. Performance drops significantly on multi-sentence inputs, particularly when using pre-trained models without re-training. These results demonstrate the necessity of SSLS for long videos.

In addition, the existing MMLMs like Gemini and Qwen2-VL perform well on video-text tasks, but they are not suitable for SSLS. Our task involves long sign language videos with thousands of frames and requires frame-level boundary detection, which inevitably results in substantial memory overhead. Therefore, we propose a task-specific model that reduces memory consumption and maintains segmentation accuracy.

7 Conclusion

In this paper, we introduce a new task, Sentence-level Sign Language Segmentation (SSLS), which aims to segment long sign language videos into non-overlapping sentence-level segments. This task facilitates the construction of large-scale sentence-level SL datasets while reducing the cost of manual annotation. To tackle this challenging problem, we first formalize SSLS as a frame-wise binary classification task and introduce a frame annotation strategy to construct dataset. Then, we design a segmentation framework to learn semantic boundaries in continuous sign language videos. Finally, we propose a boundary optimization module to mitigate over-segmentation and under-segmentation issues. Extensive experiments demonstrate the superiority of the proposed method.

Acknowledgments

This work is supported in part by National Natural Science Foundation of China under Grant No. 62172208; the project of Frontier Technologies RD Program of Jiangsu under Grant No. BF2024071; the 2024 Youth Talent Support Project of the Jiangsu Association for Science and Technology. This work is partially supported by Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

[1] Hyemin Ahn and Dongheui Lee. 2021. Refining action segmentation with hierarchical video representations. In *Proceedings of the IEEE/CVF international conference on computer vision*. 16302–16310.

[2] Samuel Albanie, Gül Varol, Liliane Momeni, Triantafyllos Afouras, Joon Son Chung, Neil Fox, and Andrew Zisserman. 2020. BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*. Springer, 35–53.

[3] Nadine Behrmann, S Alireza Golestaneh, Zico Kolter, Juergen Gall, and Mehdi Noroozi. 2022. Unified fully and timestamp supervised temporal action segmentation via sequence to sequence translation. In *European conference on computer vision*. Springer, 52–68.

[4] Yizhak Ben-Shabat, Tamar Avraham, Michael Lindenbaum, and Anath Fischer. 2018. Graph based over-segmentation methods for 3d point clouds. *Computer Vision and Image Understanding* 174 (2018), 12–23.

[5] Hannah Bull, Triantafyllos Afouras, Gül Varol, Samuel Albanie, Liliane Momeni, and Andrew Zisserman. 2021. Aligning subtitles in sign language videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 11552–11561.

[6] Hannah Bull, Michèle Gouiffès, and Annelies Braffort. 2020. Automatic segmentation of sign language into subtitle-units. In *European Conference on Computer Vision*. Springer, 186–198.

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

[8] Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro-i Nieto. 2021. How2sign: a large-scale multimodal dataset for continuous american sign language. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2735–2744.

[9] Yazan Abu Farha and Jurgen Gall. 2019. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 3575–3584.

[10] Xiao Fu, Wei Xi, Jie Yang, Yutao Bai, Zhao Yang, Rui Jiang, LI XIZHE, Jiankang Gao, and Jizhong Zhao. [n. d.]. Balanced Multimodal Learning: An Integrated Framework for Multi-Task Learning in Audio-Visual Fusion. ([n. d.]).

[11] Shiwei Gan, Yafeng Yin, Zhiwei Jiang, Hongkai Wen, Lei Xie, and Sanglu Lu. 2024. SignGraph: A Sign Sequence is Worth Graphs of Nodes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13470–13479.

[12] Shiwei Gan, Yafeng Yin, Zhiwei Jiang, Lei Xie, and Sanglu Lu. 2021. Skeleton-aware neural sign language translation. In *Proceedings of the 29th ACM International Conference on Multimedia*. 4353–4361.

[13] Daiheng Gao, Shilin Lu, Shaw Walters, Wenbo Zhou, Jiaming Chu, Jie Zhang, Bang Zhang, Mengxi Jia, Jian Zhao, Zhaoxin Fan, et al. 2024. EraseAnything: Enabling Concept Erasure in Rectified Flow Transformers. *arXiv preprint arXiv:2412.20413* (2024).

[14] Alex Graves and Alex Graves. 2012. Connectionist temporal classification. *Supervised sequence labelling with recurrent neural networks* (2012), 61–93.

[15] Kirsti Grobel and Marcell Assan. 1997. Isolated sign language recognition using hidden Markov models. In *1997 IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation*, Vol. 1. IEEE, 162–167.

[16] Dan Guo, Wengang Zhou, Meng Wang, and Houqiang Li. 2016. Sign language recognition based on adaptive hmms with data augmentation. In *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2876–2880.

[17] Lianyu Hu, Liqing Gao, Zekang Liu, and Wei Feng. 2023. Continuous sign language recognition with correlation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2529–2539.

[18] Yifei Huang, Yusuke Sugano, and Yoichi Sato. 2020. Improving action segmentation via graph-based temporal reasoning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 14024–14034.

[19] Yuchi Ishikawa, Seito Kasai, Yoshimitsu Aoki, and Hirokatsu Kataoka. 2021. Alleviating over-segmentation errors by detecting action boundaries. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*. 2322–2331.

[20] Borui Jiang, Yang Jin, Zhentao Tan, and Yadong Mu. 2023. Video action segmentation via contextually refined temporal keypoints. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 13836–13845.

[21] Peiqi Jiao, Yuecong Min, and Xilin Chen. 2025. Visual Alignment Pre-training for Sign Language Translation. In *European Conference on Computer Vision*. Springer, 349–367.

[22] Oscar Koller, Sepehr Zargaran, and Hermann Ney. 2017. Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent CNN-HMMs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4297–4305.

[23] Mikhail V Koroteev. 2021. BERT: a review of applications in natural language processing and understanding. *arXiv preprint arXiv:2103.11943* (2021).

[24] Hilde Kuehne, Alexander Richard, and Juergen Gall. 2017. Weakly supervised learning of actions from transcripts. *Computer Vision and Image Understanding* 163 (2017), 78–89.

[25] Hilde Kuehne, Alexander Richard, and Juergen Gall. 2018. A hybrid RNN-HMM approach for weakly supervised temporal action segmentation. *IEEE transactions on pattern analysis and machine intelligence* 42, 4 (2018), 765–779.

[26] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. 2017. Temporal convolutional networks for action segmentation and detection. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 156–165.

[27] Colin Lea, Rene Vidal, Austin Reiter, and Gregory D Hager. 2016. Temporal convolutional networks: A unified approach to action segmentation. In *Computer vision–ECCV 2016 workshops: Amsterdam, the Netherlands, October 8-10 and 15-16, 2016, proceedings, part III 14*. Springer, 47–54.

[28] Jun Li, Peng Lei, and Sinisa Todorovic. 2019. Weakly supervised energy-based learning for action segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*. 6243–6251.

[29] Leyang Li, Shilin Lu, Yan Ren, and Adams Wai-Kin Kong. 2025. Set you straight: Auto-steering denoising trajectories to sidestep unwanted concepts. *arXiv preprint arXiv:2504.12782* (2025).

[30] Shijie Li, Yazan Abu Farha, Yun Liu, Ming-Ming Cheng, and Juergen Gall. 2020. Ms-tcn++: Multi-stage temporal convolutional network for action segmentation. *IEEE transactions on pattern analysis and machine intelligence* 45, 6 (2020), 6647–6658.

[31] Daochang Liu, Qiyue Li, Anh-Dung Dinh, Tingting Jiang, Mubarak Shah, and Chang Xu. 2023. Diffusion action segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*. 10139–10149.

[32] Yang Liu, Jiayu Huo, Jingjing Peng, Rachel Sparks, Prokar Dasgupta, Alejandro Granados, and Sebastien Ourselin. 2023. Skit: a fast key information video transformer for online surgical phase recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 21074–21084.

[33] Shilin Lu, Yanzhu Liu, and Adams Wai-Kin Kong. 2023. Tf-icon: Diffusion-based training-free cross-domain image composition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2294–2305.

[34] Shilin Lu, Zilan Wang, Leyang Li, Yanzhu Liu, and Adams Wai-Kin Kong. 2024. Mace: Mass concept erasure in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6430–6440.

[35] Shilin Lu, Zihan Zhou, Jiayou Lu, Yuanzhi Zhu, and Adams Wai-Kin Kong. 2024. Robust watermarking using generative priors against image editing: From benchmarking to advances. *arXiv preprint arXiv:2410.18775* (2024).

[36] Zijia Lu and Ehsan Elhamifar. 2021. Weakly-supervised action segmentation and alignment via transcript-aware union-of-subspaces learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8085–8095.

[37] Zijia Lu and Ehsan Elhamifar. 2024. Fact: Frame-action cross-attention temporal modeling for efficient action segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18175–18185.

[38] Yuecong Min, Aiming Hao, Xiujuan Chai, and Xilin Chen. 2021. Visual alignment constraint for continuous sign language recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*. 11542–11551.

[39] Liliane Momeni, Gul Varol, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. 2020. Watch, read and lookup: learning to spot signs from multiple supervisors. In *Proceedings of the Asian Conference on Computer Vision*.

[40] Amit Moryossef, Zifan Jiang, Mathias Müller, Sarah Ebling, and Yoav Goldberg. 2023. Linguistically motivated sign language segmentation. *arXiv preprint arXiv:2310.13960* (2023).

[41] Ilias Papastratis, Kosmas Dimitropoulos, Dimitrios Konstantinidis, and Petros Daras. 2020. Continuous sign language recognition through cross-modal alignment of video and text embeddings in a joint-latent space. *IEEE Access* 8 (2020), 91170–91180.

[42] Xin Shen, Shaozu Yuan, Hongwei Sheng, Heming Du, and Xin Yu. 2024. Auslan-daily: Australian sign language translation for daily communication and news. *Advances in Neural Information Processing Systems* 36 (2024).

[43] Yuhan Shen and Ehsan Elhamifar. 2024. Progress-aware online action segmentation for egocentric procedural task videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18186–18197.

[44] Yuhao Su and Ehsan Elhamifar. [n. d.]. Two-Stage Active Learning for Efficient Temporal Action Segmentation. ([n. d.]).

[45] Andrés Troya-Galvis, Pierre Gançarski, Nicolas Passat, and Laure Berti-Equille. 2015. Unsupervised quantification of under-and over-segmentation for object-based remote sensing image analysis. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 8, 5 (2015), 1936–1945.

[46] Gul Varol, Liliane Momeni, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. 2021. Read and attend: Temporal localisation in sign language videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16857–16866.

[47] Xianyuan Wang, Zhenjiang Miao, Ruyi Zhang, and Shanshan Hao. 2019. I3d-lstm: A new model for human action recognition. In *IOP conference series: materials science and engineering*, Vol. 569. IOP Publishing, 032035.

[48] Fangyun Wei and Yutong Chen. 2023. Improving continuous sign language recognition with cross-lingual signs. In *Proceedings of the IEEE/CVF International*

Conference on Computer Vision. 23612–23621.

[49] Angchi Xu and Wei-Shi Zheng. 2024. Efficient and Effective Weakly-Supervised Action Segmentation via Action-Transition-Aware Boundary Alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18253–18262.

[50] Huijie Yao, Wengang Zhou, Hao Feng, Hezhen Hu, Hao Zhou, and Houqiang Li. 2023. Sign language translation with iterative prototype. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 15592–15601.

[51] Hamidullah Yasser, Josef Genabith, and Cristina España-Bonet. 2024. Sign Language Translation with Sentence Embedding Supervision. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 425–434.

[52] Jinhui Ye, Wenxiang Jiao, Xing Wang, Zhaopeng Tu, and Hui Xiong. 2023. Cross-modality data augmentation for end-to-end sign language translation. *arXiv preprint arXiv:2305.11096* (2023).

[53] Jinhui Ye, Xing Wang, Wenxiang Jiao, Junwei Liang, and Hui Xiong. 2024. Improving Gloss-free Sign Language Translation by Reducing Representation Density. *arXiv preprint arXiv:2405.14312* (2024).

[54] Fangqiu Yi, Hongyu Wen, and Tingting Jiang. 2021. Asformer: Transformer for action segmentation. *arXiv preprint arXiv:2110.08568* (2021).

[55] Huaiwen Zhang, Zihang Guo, Yang Yang, Xin Liu, and De Hu. 2023. C2st: Cross-modal contextualized sequence transduction for continuous sign language recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 21053–21062.

[56] Jihai Zhang, Wengang Zhou, and Houqiang Li. 2014. A threshold-based hmm-dtw approach for continuous sign language recognition. In *Proceedings of international conference on internet multimedia computing and service*. 237–240.

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

[58] Ronglai Zuo, Fangyun Wei, Zenggui Chen, Brian Mak, Jiaolong Yang, and Xin Tong. 2025. A simple baseline for spoken language to sign language translation with 3d avatars. In *European Conference on Computer Vision*. Springer, 36–54.