CorrNet+: Sign Language Recognition and Translation via Spatial-Temporal Correlation

Lianyu Hu, Wei Feng†, *Member, IEEE*, Liqing Gao, Zekang Liu, Liang Wan†, *Member, IEEE*

***Abstract*—In sign language, the conveyance of human body trajectories predominantly relies upon the coordinated movements of hands and facial expressions across successive frames. Despite the recent impressive advancements of sign language understanding methods, they often solely focus on individual frames, inevitably overlooking the inter-frame correlations that are essential for effectively modeling human body trajectories. To address this limitation, this paper introduces a spatial-temporal correlation network, denoted as CorrNet+, which explicitly identifies and captures body trajectories across multiple frames. In specific, CorrNet+ employs two parallel modules to build human body trajectories: a correlation module and an identification module. The former captures the cross-spacetime correlations in local spatial-temporal neighborhoods, while the latter dynamically constructs human body trajectories by distinguishing informative spatial regions. Afterwards, a temporal attention module is followed to adaptively evaluate the contributions of different frames in the whole video. The resultant features offer a holistic perspective on human body movements, facilitating a deeper understanding of sign language. As a unified model, CorrNet+ achieves new state-of-the-art performance on two extensive sign language understanding tasks, including continuous sign language recognition (CSLR) and sign language translation (SLT). Especially, CorrNet+ surpasses previous methods equipped with resource-intensive pose-estimation networks or pre-extracted heatmaps for hand and facial feature extraction. Compared with CorrNet, CorrNet+ achieves a significant performance boost across all benchmarks while halving the computational overhead, achieving a better computation-accuracy trade-off. A comprehensive comparison with previous spatial-temporal reasoning methods verifies the superiority of CorrNet+. Code is available at https://github.com/hulianyuyy/CorrNet_Plus.**

***Index Terms*—Continuous sign language recognition, Sign language translation, Spatial-temporal correlation, Model efficiency.**

I. Introduction

Sign language is one of the most widely-used communication tools for the deaf community in their daily life, which mainly conveys its meaning by facial expressions, head movements, hand gestures and body postures [1], [2]. However, mastering this language remains an overwhelming challenge for the hearing people, thus hindering direct interactions between two distinct groups. To alleviate this barrier, recent strides in automatic sign language understanding techniques [3], [4] have emerged, broadly categorized into three distinct domains: (1) isolated sign language recognition (ISLR), which aims to classify a video segment into an independent gloss[1]; (2) continuous sign language recognition (CSLR), which progresses by classifying the input sign videos into a series of glosses to express sentences, instead of recognizing a single gloss only; (3) sign language translation (SLT), which directly translating the input sign videos into spoken texts that can be naturally understood by the hearing people. The difference of these tasks is illustrated in Fig. 1(a). To hopefully bridge the communication gaps between two groups, this paper focuses on CSLR and SLT, as they hold greater promise for real-life applications in sign language systems.

Evidently, human body trajectories serve as prominent cues for understanding actions in human-centric video comprehension, which have gained substantial attention across various tasks [5]–[10]. In sign language, these trajectories are mainly conveyed by both manual components (hand/arm gestures), and non-manual components (facial expressions, head movements, and body postures) [1], [2]. Especially, the coordinated horizontal and vertical movements of human face and both hands, coupled with adjoint actions like finger twisting and facial expressions, play a major role in expressing sign language. Tracking and leveraging the trajectories of these crucial body parts is of great benefit to understanding sign language.

However, current sign language methods [12]–[21] usually treat each frame equally, overlooking their cross-frame interactions and thereby failing to leverage human body trajectories. Especially, they usually adopt a shared 2D CNN to independently extract spatial features for each frame [12], [15], [18], [20], [21]. Consequently, frames are processed individually without considering their interactions, thus inhibited to harness the potential of cross-frame trajectories for sign comprehension. Some methods propose to use a 3D or (2+1)D CNN [13], [22] to capture the local cross-spacetime features. However, their fixed design and limited spatial-temporal receptive fields hinder the establishment of spatial relationships across distant regions. Moreover, these methods incur substantial computational costs compared to their 2D counterparts. Alternative temporal techniques, such as temporal shift [23] or temporal convolutions [24], could address short-term temporal dynamics. However, it's hard for them to aggregate information from distant spatial regions due to the limited spatial-temporal receptive field. Besides, they may fail to dynamically model human body movements for different samples with a fixed structure during inference. With the above considerations, it's necessary to develop an effective and efficient method for capturing human body trajectories to advance sign language comprehension.

Lianyu Hu, Wei Feng, Liqing Gao, Zekang Liu, Liang Wan are with the College of Intelligence and Computing, Tianjin University, Tianjin 300350, China (e-mail: hly2021@tju.edu.cn; wfeng@ieee.org; lwan@tju.edu.cn).

† Wei Feng and Liang Wan are the Corresponding authors.

[1]Gloss is the atomic lexical unit to annotate sign languages.

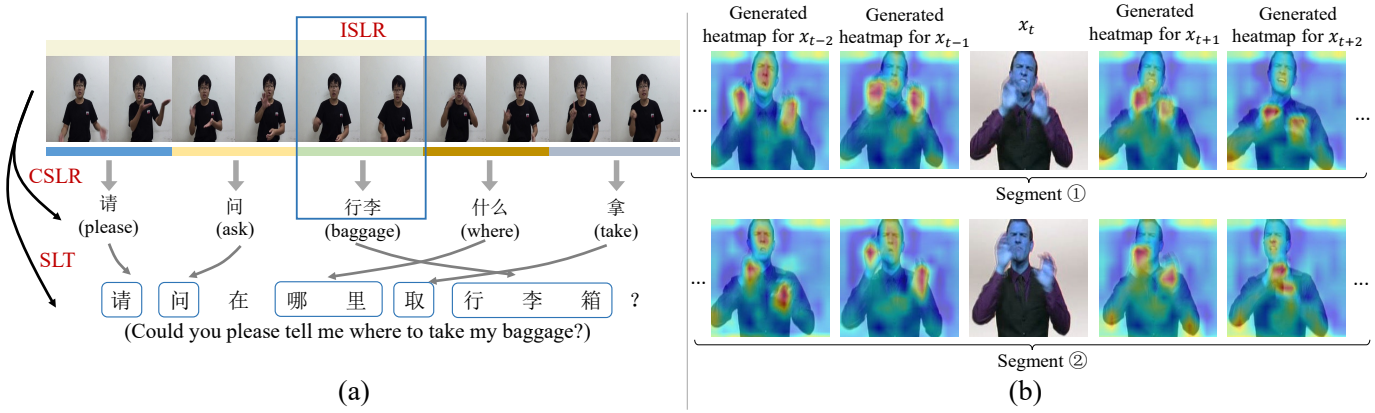


Fig. 1. (a) Illustration for the difference among the isolated sign language recognition (ISLR) task, continuous sign language recognition (CSLR) task and sign language translation (SLT) task. (b) Visualization of correlation maps with Grad-CAM [11] between the current frame and two adjacent frames in the left/right side. It's observed that without extra supervision, our method well attends to informative regions in adjacent frames to identify human body trajectories.

To address these challenges, we introduce CorrNet+, a novel framework explicitly designed to model human body trajectories across adjacent frames. As depicted in Figure 1(b), our approach dynamically attends to the movements of informative regions across wide spatial distances. Unlike certain prior methods [16], [22], [25], [26] that rely on expensive supervision such as pose estimation techniques or body heatmaps, our method alleviates the need for such resource-intensive guidance and can be trained in a self-motivated manner. Notably, our approach achieves superior performance compared to previous methods while significantly reducing the required computational demands.

CorrNet+ employs two parallel modules to build human body trajectories: a correlation module and an identification module. The former computes correlation maps within a local spatial-temporal region to identify human body trajectories. The latter dynamically emphasizes the informative regions that convey critical information. Besides these two components, considering human body trajectories are unevenly distributed in the video, a temporal attention module is then introduced to highlight the critical human body movements. The generated features provide a comprehensive perspective on human body movements, thereby enhancing the comprehension of sign language. Remarkably, CorrNet+ achieves new state-of-the-art performance on three large-scale CSLR benchmarks (PHOENIX2014 [27], PHOENIX2014-T [28] and CSL-Daily [29]), and two widely-used SLT benchmarks (PHOENIX2014-T [28] and CSL-Daily [29]). Especially, CorrNet+ largely outperforms previous methods equipped with resource-intensive pose-estimation networks or pre-extracted heatmaps for hand and facial feature extraction [16], [22], [25], [26]. Compared with CorrNet [30], CorrNet+ brings notable performance gain across all benchmarks and drastically reduces the consumed computations by half, achieving a better computation-accuracy trade-off. A comprehensive comparison with other spatial-temporal reasoning methods demonstrates the superiority of CorrNet+. Visualizations hopefully verify the efficacy of CorrNet+ on emphasizing human body trajectories across adjacent frames. Abundant ablations demonstrate the effects of each component within CorrNet+.

This paper is a substantial extension from a preliminary conference version [30] with a number of major changes. First, we reformulate the design of the correlation module in Section 3.2 to make it more lightweight and powerful, which is a key component for effectively modeling human body trajectories. Second, a new temporal attention module is introduced to dynamically emphasize the critical body trajectories in Section 3.4. Finally, we incorporate new results on the SLT benchmarks, and significantly extend the experimental results on the CSLR benchmarks in Section 4. We additionally append new visualizations to clearly show the effects of our proposed method. The remainder of this paper is organized as follows. Section 2 reviews the related work. Section 3 elaborates the proposed method. Section 4 reports the experimental results, followed by a brief conclusion in Section 5.

II. Related Work

A. Continuous Sign Language Recognition

Continuous sign language recognition tries to translate image frames into corresponding glosses in a weakly-supervised way: only sentence-level label is provided. Earlier methods [31], [32] in CSLR always employ hand-crafted features or HMM-based systems [27], [33]–[35] to perform temporal modeling and translate sentences step by step. Hand-crafted features [31], [32] are carefully selected to provide better visual information. HMM-based systems [27], [33]–[35] first employ a feature extractor to capture visual features and then adopt an HMM to perform long-term temporal modeling.

The recent success of convolutional neural networks (CNNs) and recurrent neural networks (RNNs) brings huge progress for CSLR. The widely used CTC loss [36] in recent CSLR methods [12]–[15], [37], [38] enables training deep networks in an end-to-end manner by sequentially aligning target sentences with input frames. These CTC-based methods first rely on a feature extractor, i.e., 3D or 2D&1D CNN hybrids, to extract frame-wise features, and then adopt a LSTM for capturing long-term temporal dependencies. However, several methods [13], [37] found in such conditions the feature extractor is not well-trained and then present an

iterative training strategy to relieve this problem, but consume much more computations. Some recent studies [12], [15], [17], [39] try to directly enhance the feature extractor by adding alignment losses [15], [17], [39] or adopt pseudo labels [12] in a lightweight way, alleviating the heavy computational burden. TLP [40] proposes to enhance the temporal information extraction process by designing advanced temporal pooling methods. SEN [18] tries to locate the informative spatial regions in sign videos in a self-supervised way. CVT-SLR [21] employs a contrastive visual-textual transformation to tackle the insufficient training problem existed in CSLR. CTCA [20] designs a cross-temporal context aggregation module to enhance local temporal context and global temporal context.

Our method is designed to explicitly incorporate body trajectories to identify a sign, especially those from hands and face. Some previous methods have also explicitly leveraged the hand and face features for better recognition. For example, CNN-LSTM-HMM [26] employs a multi-stream HMM (including hands and face) to integrate multiple visual inputs to improve recognition accuracy. STMC [25] first utilizes a pose-estimation network to estimate human body keypoints and then sends cropped appearance regions (including hands and face) for information integration. C^2SLR [16] leverages the pre-extracted pose keypoints as supervision to guide the model to explicitly focus on hand and face regions. TwoStream Network [22] builds two branches consisting of a visual branch and a pose branch to fuse beneficial information from complementary modalities. Our method doesn't rely on additional cues like heavy pose estimation networks [16], [22], [25] or multiple streams [26] which consume much more computations to leverage hand and face information. Instead, our model could be end-to-end trained to dynamically attend to body trajectories in a self-motivated and lightweight way.

B. Sign Language Translation

Camgoz et al. [28] pioneer the neural SLT task and publish the neural dataset PHOENIX2014-T [28] which regards the SLT as a sequence-to-sequence problem. They implement the neural SLT system using the encoder-decoder paradigm [41]. This paradigm is adopted by subsequent studies which focus on addressing the challenges of data scarcity and domain gap. Then, SLRT [42] first introduces a Transformer-based encoder-decoder framework to perform end-to-end SLT, with a Connectionist Temporal Classification (CTC) loss [36] to soft-match sign representations and gloss sequences. STMC-T [25] improves sign language translation by introducing multiple cues aimed by a pose estimation network. SignBack [29] tries to handle the insufficient training data problem by introducing back-translation techniques to generate new pseudo samples. Motivated by the progress of neural machine translation (NMT), several methods attempt to introduce these advanced techniques into SLT. For example, Chen et al. [22], [43] made the first attempt to introduce large language models into SLT with carefully designed pretraining strategies. XmDA [44] presents two new data augmentation methods, namely, cross-modality mix-up and cross-modality knowledge distillation to expand the training samples. Zhu el al. [45] testifies the effectiveness of several NMT techniques including data augmentation, transfer learning and multilingual NMT on SLT. Most existing methods adopt gloss representations as an intermediate state to promote translation accuracy. Some methods [19], [46] propose to eliminate the need of label-laboring glosses and design gloss-free SLT methods. GloFE [46] presents an end-to-end sign language translation framework by exploiting the shared underlying semantics of signs and the corresponding spoken translation. GFSLT-VLP [19] improves SLT by inheriting language-oriented prior knowledge from pretrained models, without any gloss annotation assistance.

C. Applications of Correlation Operation

Correlation operation has been widely used in various domains, especially optical flow estimation and video action recognition. Rocco et al. [47] used it to estimate the geometric transformation between two images, and Feichtenhofer et al. [48] applied it to capture object co-occurrences across time in tracking. For optical flow estimation, Deep matching [49] computes the correlation maps between image patches to find their dense correspondences. CNN-based methods like FlowNet [50] and PWC-Net [51] design a correlation layer to help perform multiplicative patch comparisons between two feature maps. More recently, VideoFlow [52] proposes to propagate motion correlations between adjacent frames for multi-frame optical flow estimation. FlowFormer++ [53] introduces a masked autoencoding pretraining strategy and encodes the cross-frame correlations to help optical flow estimation. For video action recognition, Zhao et al. [54] firstly employ a correlation layer to compute a cost volume to estimate the motion information. STCNet [55] considers spatial correlations and temporal correlations, respectively, inspired by SENet [56]. MFNet [57] explicitly estimates the approximation of optical flow based on fixed motion filters. Wang et al. [58] design a learnable correlation filter and replace 3D convolutions with the proposed filter to capture spatial-temporal information. PCD [59] presents to minimize the distribution of correlation information in videos for domain adaptation. Different from these methods that explicitly or implicitly estimate optical flow, the correlation operator in our method is used in combination with other operations to identify and track body trajectories across frames.

III. METHOD

A. Overview

As shown in Fig. 2, our model comprises a foundational base model, followed by different task-specific heads to support various sign language understanding tasks. Given a sign video with T input frames $\boldsymbol{x} = \{\boldsymbol{x}_t^0\}_{t=1}^T \in \mathcal{R}^{T\times 3\times H_0\times W_0}$ with spatial size of $H_0 \times W_0$, the base model first uses a feature extractor instantiated as a 2D CNN[2] to extract spatial-wise features $\boldsymbol{v} = \{\boldsymbol{v}_t\}_{t=1}^T \in \mathcal{R}^{T\times d}$ with d representing the number of channels. It further incorporates a 1D CNN and a

[2]Here we only consider the feature extractor based on 2D CNN, because recent findings [3], [16] show 3D CNN can not provide as precise gloss boundaries as 2D CNN, and lead to lower accuracy.

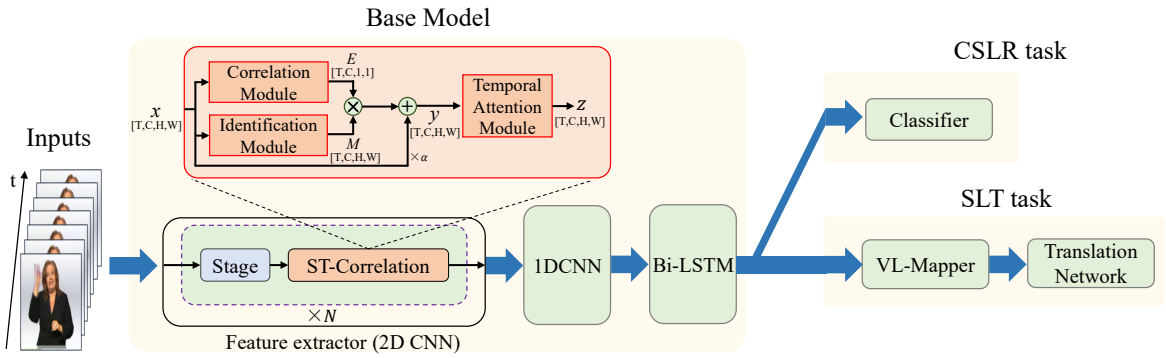


Fig. 2. An overview for our CorrNet+, which can support both the CSLR task and the SLT task with a common base model. In this base model, it first employs a feature extractor (2D CNN) to capture frame-wise features, and then adopts a 1D CNN and a BiLSTM to perform short-term and long-term temporal modeling, respectively. For the CSLR task, we attach a classifier instantiated as a fully connected layer to perform classification. For the SLT task, we attach a VL-mapper instantiated as a MLP and a translation network to predict sentences. The feature extractor is consisted of multiple stages to extract spatial-wise features for each frame independently. After each stage of the feature extractor, we insert a correlation stage to capture cross-frame interactions. An identification module and a correlation module are first concurrently placed to identify body trajectories across adjacent frames, whose outputs are then element-wisely multiplied and fed into the temporal attention module to dynamically emphasize the key human body trajectories in the whole video.

BiLSTM to perform short-term and long-term temporal modeling, respectively. Various task-specific heads are attached to support different sign language understanding tasks. For the CSLR task, we attach a classifier instantiated as a fully connected layer to recognize the input video into a series of glosses $\boldsymbol{g} = \{\boldsymbol{g}_i\}_{i=1}^{N}$. Here, N denotes the length of the label sequence. This process is supervised by the widely-used CTC loss [36] $\mathcal{L}_{\text{CTC}}$ to align input video frames with target gloss sequences. For the SLT task, we attach a visual-language (VL) mapper instantiated as a MLP and a translation network to translate the gloss-wise features v into spoken texts $\boldsymbol{s} = \{\boldsymbol{s}_i\}_{i=1}^{H}$. Here, H denotes the length of the output text sequence. This procedure is supervised by the standard sequence-to-sequence cross-entropy loss [60] $\mathcal{L}_{\text{CE}}$.

Despite the recent advancements in sign language understanding methods, they usually treat each frame equally by using a common 2D CNN to extract spatial-wise features and thus fail to capture cross-frame interactions. While some methods propose to model local spatial-temporal information with spatial-temporal reasoning methods like 3D CNN [13], [22] and temporal convolutions, they suffer from excessive computations and limited spatial-temporal receptive fields. Consequently, they struggle to effectively capture human body movements across a broader spatial-temporal region. To address these limitations, we design a spatial-temporal correlation network (CorrNet+) as shown in Fig. 2. We seamlessly insert a spatial-temporal correlation network (ST-correlation) after each stage in the feature extractor to capture the local spatial-temporal correlations for each frame. Specifically, We simultaneously deploy two critical components including a correlation module and an identification module to capture the cross-frame interactions and identify informative spatial regions. The outputs $\boldsymbol{E}$ and $\boldsymbol{M}$ from both modules are element-wisely multiplied and then added via a residual input connection, yielding intermediate representations $\boldsymbol{y}$. We then feed $\boldsymbol{y}$ into a temporal attention module to dynamically evaluate the contributions of different frames in the whole video to emphasize keyframes and suppress meaningless ones. We next introduce each component in detail.

B. Correlation Module

As a rich and expressive communication protocol, sign language is mainly conveyed by both manual components (hand/arm gestures), and non-manual components (facial expressions, head movements, and body postures) [1], [2]. However, these informative body parts, e.g., hands and face, often exhibit misalignment across adjacent frames. To address this spatial discrepancy and establish connections between distant spatial regions, we propose a novel approach by computing correlation maps between neighboring frames to identify and track human body trajectories. We first briefly recap the solution of CorrNet [30] and naturally introduce our solution to overcome its inherent limitations.

Formally, each frame could be represented as a 3D tensor $\boldsymbol{x}_t \in \mathcal{R}^{C\times H\times W}$, where C represents the number of channels and $H \times W$ denotes spatial size. In CorrNet [30], we compute the affinities between all patches in the current frame $\boldsymbol{x}_t$ and patches in adjacent frames to model human body trajectories. Taking a feature patch $\boldsymbol{p}_t(i,j)$ with the spatial location (i,j) in the current frame $\boldsymbol{x}_t$ as an example, its affinity $\boldsymbol{A}(i,j,i',j')$ with another patch $\boldsymbol{p}_{t+1}(i',j')$ in $\boldsymbol{x}_{t+1}$ is computed in a dot-product way as:

$$\boldsymbol{A}(i,j,i',j') = \frac{1}{C}\sum_{c=1}^{C} \boldsymbol{p}_t^c(i,j) \times \boldsymbol{p}_{t+1}^c(i',j'). \tag{1}$$

Fig. 3(a) illustrates this process. However, the computed correlation maps yield a tensor of size $H \times W \times H \times W$, resulting in an overall computation complexity of $O(H^2W^2)$ quadratic to the number of patches. Though this operation can effectively build cross-frame interactions to handle the spatial misalignment, it imposes a substantial computational burden. Moreover, the high computational costs restrict the spatial-temporal interactions to neighboring frames, hindering our ability to consecutively capture human body trajectories across a broader temporal context.

To handle these limitations, we reformulate the correlation module to make it more lightweight and powerful, whose framework is shown in fig. 4. Specifically, we compress all

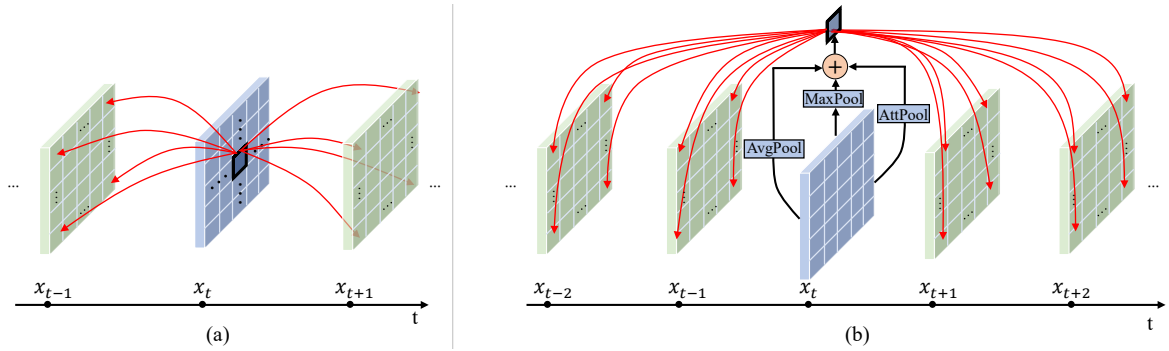


Fig. 3. Illustration for the difference between the correlation operator in CorrNet [30] and CorrNet+. (a) CorrNet [30]. It computes correlation maps between a spatial patch $p_t(i,j)$ in x_t and all other patches in adjacent frame x_{t+1} and x_{t-1}. The overall computation complexity is $O(H^2W^2)$, quadratic to the number of spatial patches in each frame, which incurs heavy extra computations. (b) To reduce computations, we condense the features of x_t into several compact representations, which are then used to compute correlation maps with adjacent frames on behalf of x_t. In this case, as the number of selected patches is reduced from $H \times W$ to $O(1)$ for x_t, the computation complexity is drastically decreased from $O(H^2W^2)$ to $O(HW)$. It also enables us to compute correlation maps with neighbors in a larger temporal duration to more effectively capture the whole human body movements in expressing a sign.

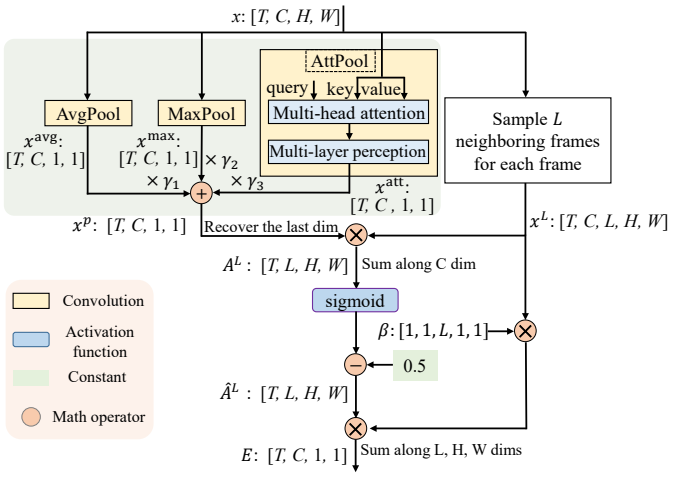


Fig. 4. An framework overview for our proposed correlation module. It first condenses each frame into a compact representation, and then uses it to compute correlation maps with adjacent frames within a predefined range of L to model human body trajectories.

patches in $\boldsymbol{x}_t$ into a compact tensor to compute the correlation maps with significantly reduced computational overhead. We further extend the spatial-temporal neighborhood of the correlation operator to capture the trajectories of the signer in a large temporal duration.

In specific, we use three different ways to compress the features of each frame from various views. For simplicity, we choose the average aggregation, maximum aggregation and attention aggregation functions as our protocols.

For average aggregation, given the input feature $\boldsymbol{x} \in \mathcal{R}^{T\times C\times H\times W}$, we perform average pooling along the spatial dimension to transform it into a representation $\boldsymbol{x}^{\text{avg}} \in \mathcal{R}^{T\times C\times 1\times 1}$ as:

$$\boldsymbol{x}^{\text{avg}} = \text{AvgPool}(\boldsymbol{x}). \tag{2}$$

For maximum aggregation, we perform max pooling to compress $\boldsymbol{x}$ into a representation $\boldsymbol{x}^{\text{max}} \in \mathcal{R}^{T\times C\times 1\times 1}$ as:

$$\boldsymbol{x}^{\text{max}} = \text{MaxPool}(\boldsymbol{x}). \tag{3}$$

For attention aggregation, we randomly initialize a tensor $\boldsymbol{q} \in \mathcal{R}^{1\times C\times 1\times 1}$ acting as a query. It is then used to compute affinities $\boldsymbol{A} \in \mathcal{R}^{T\times 1\times H\times W}$ with patches in each frame following the multi-head attention (MHA) [60] process, whose features are fed into a Multi-Layer Perception (MLP) module [60] to obtain the output $\boldsymbol{x}^{\text{att}} \in \mathcal{R}^{T\times C\times 1\times 1}$ as:

$$\boldsymbol{x}^{\text{att}} = \text{MLP}(\text{MHA}(\text{query} = \boldsymbol{q}, \text{key} = \boldsymbol{x}, \text{value} = \boldsymbol{x})). \tag{4}$$

In this procedure, the number of heads is set as 1 for the MHA process, and the dimension expansion factor is 1 for the MLP module to minimize computations.

After obtaining the condensed features $\boldsymbol{x}^{\text{avg}}$, $\boldsymbol{x}^{\text{max}}$ and $\boldsymbol{x}^{\text{att}}$, we combine them into a compact representation. Practically, we multiply these features with a learnable coefficient $\boldsymbol{\gamma} \in \mathcal{R}^3$ to control their importance for fusion to obtain $\boldsymbol{x}^p \in \mathcal{R}^{T\times C\times 1\times 1}$ as:

$$\boldsymbol{x}^p = \boldsymbol{x}^{\text{avg}} \times \boldsymbol{\gamma}_1 + \boldsymbol{x}^{\text{max}} \times \boldsymbol{\gamma}_2 + \boldsymbol{x}^{\text{att}} \times \boldsymbol{\gamma}_3. \tag{5}$$

Here, $\boldsymbol{\gamma}$ is initialized as a tensor filled with values of $\frac{1}{3}$, and then updated via gradient-based backward propagation in the training process. Especially, as only one compact representation is used on behalf of the current frame, the computation complexity of calculating correlation maps between adjacent frames can be drastically reduced to only $O(HW)$, in contrast to $O(H^2W^2)$ in CorrNet [30]. In practice, the computations are notably decreased from 3.64 GFLOPs[3] to 0.01 GFLOPs, bringing only quite a few extra computations.

Considering sign language is mainly conveyed by consecutive human body motion like hand and arm movements, it's necessary to identify and track the body trajectories in a large temporal neighborhood to understand signs. We strategically enlarge the temporal receptive field of the correlation module to achieve this goal. Specifically, for an input video $\boldsymbol{x} \in \mathcal{R}^{T\times C\times H\times W}$, we sample L neighboring frames for each frame to formulate a neighboring frame set $\boldsymbol{x}^L \in \mathcal{R}^{T\times C\times L\times H\times W}$. We then recover the last dimension of $\boldsymbol{x}^p$ as $\boldsymbol{x}^p \in \mathcal{R}^{T\times C\times 1\times 1\times 1}$ and use it to compute affinities

[3]FLOPs denotes the number of multiply-add operations and GFLOPs denotes measuring FLOPs by giga.

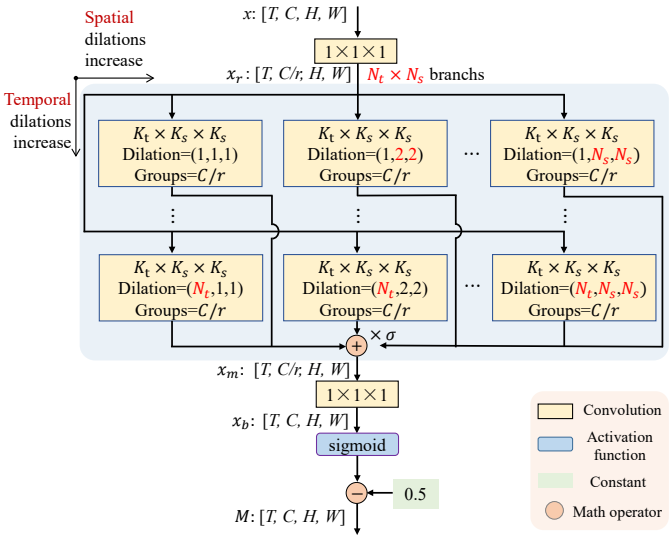


Fig. 5. Illustration for our identification module. To avoid heavy computations in identifying informative spatial regions when modeling local spatial-temporal information, we decompose the spatial-temporal modeling structure along the spatial and temporal dimensions simultaneously to form a multiscale architecture, enlarging the model capacity.

with $\boldsymbol{x}^L$ to obtain the local spatial-temporal correlation maps $\boldsymbol{A}^L \in \mathcal{R}^{T\times L\times H\times W}$ as:

$$\boldsymbol{A}^L = \sum_{i=0}^{C} \boldsymbol{x}^p_{:i} \times \boldsymbol{x}^L_{:i} \tag{6}$$

where : denotes taking all elements in the corresponding dimension. L can be set as various values in different network stages to capture information of different temporal scales.

Given the spatial-temporal correlation maps $\boldsymbol{A}^L$, we constrain values in $\boldsymbol{A}^L$ into the range of (0,1) by passing it through a sigmoid function. We further subtract 0.5 from the results to emphasize informative regions with positive values and suppress redundant areas with negative values as:

$$\hat{\boldsymbol{A}}^L = \text{sigmoid}(\boldsymbol{A}_L) - 0.5. \tag{7}$$

After identifying the correlations between adjacent frames, we incorporate them back into each frame to reason about the local human body movements. Specifically, we recover the second dimension of the cross-frame correlations $\hat{\boldsymbol{A}}^L$ and repeat it for C times to obtain $\hat{\boldsymbol{A}}^L \in \mathcal{R}^{T\times C\times L\times H\times W}$. We then use $\hat{\boldsymbol{A}}^L$ to multiply with the features of the neighboring frame set $\boldsymbol{x}^L$ to obtain the local human body trajectories $\boldsymbol{E} \in \mathcal{R}^{T\times C\times 1\times 1}$ as:

$$\boldsymbol{E} = \sum_{l=1}^{L} \sum_{i',j'} \hat{\boldsymbol{A}}^L_{::l}(i',j') \times \boldsymbol{x}^L_{::l}(i',j') \times \boldsymbol{\beta}_{::l} \tag{8}$$

where a learnable coefficient $\boldsymbol{\beta} \in \mathcal{R}^{1\times 1\times L\times 1\times 1}$ is attached to measure the importance of different neighboring frames. $\boldsymbol{\beta}$ is initialized as a tensor filled with values of $\frac{1}{L}$, and updated via gradient-based backward propagation in the training process. This correlation calculation is repeated for each frame in a video to track body trajectories in videos.

C. Identification Module

The correlation module computes correlation maps among spatial-temporal neighboring patches to model cross-frame interactions. However, not all regions play an equal role in sign expression. Therefore, it's critical to selectively emphasize informative regions that carry essential body trajectories within the current frame x_t and suppress background noise and non-critical elements. To achieve this goal, we present an identification module to dynamically emphasize these informative spatial regions. Specifically, as informative regions like hand and face are misaligned in adjacent frames, the identification module leverages the closely correlated local spatial-temporal features to tackle the misalignment and locate informative spatial regions.

As shown in Fig. 5, the identification module first projects input features $\boldsymbol{x} \in \mathcal{R}^{T\times C\times H\times W}$ into $\boldsymbol{x}_r \in \mathcal{R}^{T\times C/r\times H\times W}$ with a $1\times 1\times 1$ convolution to decrease the computations, with a channel reduction factor r as 16 by default.

As the informative regions, e.g., hands and face, are not exactly aligned in adjacent frames, it's necessary to consider a large spatial-temporal neighborhood to identify these features. Instead of directly employing a large 3D spatial-temporal kernel, we present a multi-scale paradigm by decomposing it into parallel branches of progressive dilation rates to reduce required computations and increase the model capacity.

Specifically, as shown in Fig. 5, with a same small base convolution kernel of $K_t \times K_s \times K_s$, we employ multiple convolutions with their dilation rates increasing along spatial and temporal dimensions concurrently. The spatial and temporal dilation rates range within (1, N_s) and (1, N_t), respectively, resulting in total $N_s \times N_t$ branches. Group-wise convolutions are employed for each branch to reduce parameters and computations. Features from different branches are multiplied with learnable coefficients $\{\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_{N_s\times N_t}\}$ to control their importance, and then added to mix information from branches of various spatial-temporal receptive fields as:

$$\boldsymbol{x}_m = \sum_{i=1}^{N_s} \sum_{j=1}^{N_t} \boldsymbol{\sigma}_{ij} \times \text{conv}^s_{ij}(\boldsymbol{x}_r) \tag{9}$$

where the group-wise convolution $\text{conv}^s_{i,j}$ of different branches receives features of different spatial-temporal neighborhoods, with dilation rate (j, i, i).

After receiving features from a large spatial-temporal neighborhood, x_m passes through a convolution with kernel size of 1 to project the features into $\boldsymbol{x}_b \in \mathcal{R}^{T\times C\times H\times W}$ to recover the channels from C/r to C. We then pass $\boldsymbol{x}_b$ through a sigmoid function to generate attention maps with values ranging within (0,1), which are further subtracted from 0.5 to obtain $\boldsymbol{M} \in \mathcal{R}^{T\times C\times H\times W}$ to emphasize informative regions with positive values and suppress redundant areas with negative values as:

$$\boldsymbol{M} = \text{sigmoid}(\text{conv}_{1\times 1\times 1}(\boldsymbol{x}_m)) - 0.5. \tag{10}$$

Given the attention maps $\boldsymbol{M}$ to identify informative regions, it's multiplied with the cross-frame interactions $\boldsymbol{E}$ computed by the correlation module to emphasize critical spatial regions

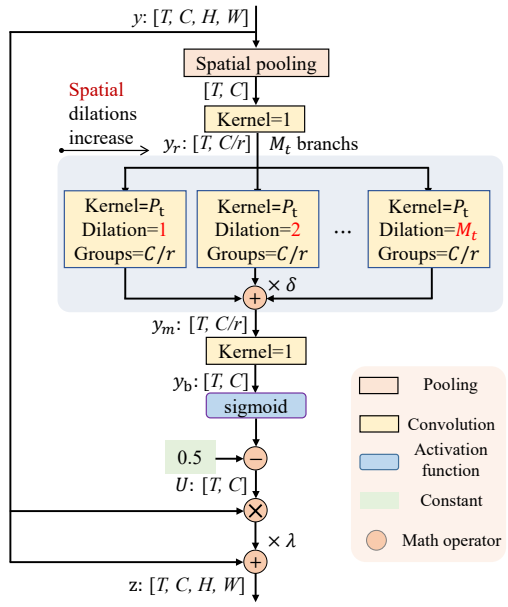


Fig. 6. Illustration for our temporal attention module. We employ a temporal multiscale architecture to aggregate local temporal information to dynamically evaluate the contributions of each frame in a lightweight manner.

that convey body trajectories and suppress others like background or noise. This refined trajectory information is finally incorporated into original spatial features $\boldsymbol{x}$ via a residual connection as:

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{\alpha}\boldsymbol{E} \times \boldsymbol{M}. \tag{11}$$

Here, $\boldsymbol{\alpha}$ is initialized as zero to keep the original spatial features and makes the model keep original behaviors.

D. Temporal Attention Module

The above modules effectively identify the critical cross-frame interactions within informative spatial regions. However, across the entire video, not all frames are equally important in expressing sign language. Some frames carry crucial information while others merely convey idle meanings. To address this, we introduce a temporal attention module. Drawing inspiration from the design principles of the identification module, we dynamically consider the importance of different frames to adaptively emphasize the keyframes and suppress others.

Fig. 6 gives the overview of the temporal attention module. Given the input features $\boldsymbol{y} \in \mathcal{R}^{T\times C\times H\times W}$ generated by the correlation module and identification module, we first perform spatial pooling to eliminate the spatial dimensions, and then project the features into $\boldsymbol{y}_r \in \mathcal{R}^{T\times C/r\times H\times W}$ with a convolution kernel size of 1 to decrease the computations.

To sufficiently evaluate the contributions of different frames, we propose a multiscale architecture to leverage the local information in a large temporal neighborhood. In specific, as shown in Fig. 6, with a same small temporal kernel of P_t, multiple parallel depth-wise convolutions are concurrently employed with different dilation rates ranging from 1 to M_t to model information from various temporal receptive fields. Features from different branches are multiplied with learnable coefficients $\{\boldsymbol{\delta}_1, \ldots, \boldsymbol{\delta}_{P_t}\}$ to adjust their importance and added to fuse complementary information from different temporal ranges as:

$$\boldsymbol{y}_m = \sum_{i=1}^{P_t} \boldsymbol{\delta}_i \times \mathrm{conv}_i^t(\boldsymbol{y}_r) \tag{12}$$

where conv_i^t denotes the group-wise convolution of i-th branch with dilation rate i.

After receiving the closely correlated spatial-temporal information, y_m passes through a convolution with kernel size of 1 to project the features into $\boldsymbol{y}_b \in \mathcal{R}^{T\times C\times H\times W}$ to recover the channels from C/r to C. We then pass $\boldsymbol{y}_b$ through a sigmoid function to generate temporal attention maps with values ranging within [0,1], which are further subtracted from 0.5 to obtain $\boldsymbol{U} \in \mathcal{R}^{T\times C}$ to emphasize keyframes with positive values and suppress others with negative values as:

$$\boldsymbol{U} = \mathrm{sigmoid}(\mathrm{conv}_1(\boldsymbol{y}_m)) - 0.5. \tag{13}$$

We then recover the spatial dimensions of $\boldsymbol{U}$ to obtain $\boldsymbol{U} \in \mathcal{R}^{T\times C\times H\times W}$, and multiply it with input features $\boldsymbol{y}$ to dynamically adjust the weights of input frames according to their contributions. These augmented representations are further incorporated into the input features $\boldsymbol{y}$ via a residual connection as:

$$\boldsymbol{z} = \boldsymbol{y} + \boldsymbol{\lambda}\boldsymbol{y} \times \boldsymbol{U}. \tag{14}$$

Here, $\boldsymbol{\lambda}$ is initialized as zero during training to avoid hurting the original temporal features.

IV. Experiments

A. Experimental Setup

1) Datasets.: **PHOENIX2014** [27] is recorded from a German weather forecast broadcast with nine actors before a clean background with a resolution of 210 × 260. It contains 6841 sentences with a vocabulary of 1295 signs, divided into 5672 training samples, 540 development (Dev) samples and 629 testing (Test) samples.

PHOENIX2014-T [28] is available for both CSLR and sign language translation tasks. It contains 8247 sentences with a vocabulary of 1085 signs, split into 7096 training instances, 519 development (Dev) instances and 642 testing (Test) instances. It can be used for both *CSLR* and *SLT* tasks.

CSL-Daily [29] revolves the daily life, recorded indoor at 30fps by 10 signers. It contains 20654 sentences, divided into 18401 training samples, 1077 development (Dev) samples and 1176 testing (Test) samples. It can be used for both *CSLR* and *SLT* tasks.

CSL [61] is collected in the laboratory by fifty signers with a vocabulary size of 178 with 100 sentences. It contains 25000 videos, divided into training and testing sets by a ratio of 8:2.

2) Training details.: For fair comparisons, we follow the same setting as state-of-the-art methods [15], [16] to prepare our model. We adopt ResNet18 [62] as the 2D CNN backbone with ImageNet [63] pretrained weights. The 1D CNN of state-of-the-art methods is set as a sequence of {K5, P2, K5, P2} layers where Kθ and Pθ denotes a 1D convolutional layer and a pooling layer with kernel size of θ, respectively. A two-layer BiLSTM with hidden size 1024 is attached for long-term

TABLE I
ABLATIONS FOR THE EFFECTIVENESS OF THE PROPOSED CORRELATION MODULE, IDENTIFICATION MODULE AND TEMPORAL ATTENTION MODULE ON THE PHOENIX2014 DATASET.

Correlation	Identification	Temporal Weighting	Dev(%)	Test(%)
✘	✘	✘	20.2	21.0
✓	✘	✘	19.2	19.7
✘	✓	✘	19.5	20.1
✘	✘	✓	19.6	20.2
✓	✓	✘	18.4	18.7
✓	✘	✓	18.8	19.2
✘	✓	✓	19.0	19.3
✓	✓	✓	**18.0**	**18.2**

TABLE II
ABLATIONS FOR THE LOCATIONS OF CORRNET+ ON THE PHOENIX2014 DATASET.

Stage 2	Stage 3	Stage 4	Dev(%)	Test(%)
✘	✘	✘	20.2	21.0
✓	✘	✘	19.3	19.9
✘	✓	✘	19.2	19.7
✘	✘	✓	19.0	19.5
✓	✓	✘	18.5	18.8
✓	✓	✓	**18.0**	**18.2**

TABLE III
ABLATIONS FOR THE EFFECTIVENESS OF CORRELATION MODULE ON THE PHOENIX2014 DATASET.

Configurations	Dev(%)	Test(%)	Extra GFLOPs/ Original GFLOPs
CorrNet [30]	18.8	19.4	3.600 / 3.640
CorrNet+	**18.0**	**18.2**	**0.010 / 3.640**
-	20.2	21.0	-
L=[2,2,2]	19.0	19.0	0.007 / 3.640
L=[6,6,6]	18.5	18.8	0.010 / 3.640
L=[10,10,10]	18.4	18.7	0.012 / 3.640
L=[2,6,10]	**18.0**	**18.2**	**0.010 / 3.640**
L=[10,6,2]	18.6	18.8	0.012 / 3.640
L=[6,10,14]	18.3	18.4	0.012 / 3.640

temporal modeling, followed by a fully connected layer for sentence prediction. We train our models for 80 epochs with initial learning rate 0.001 which is divided by 5 at epoch 40 and 60. Adam [64] optimizer is adopted as default with weight decay 0.0001 and batch size 2. All input frames are first resized to 256×256, and then randomly cropped to 224×224 with 50% horizontal flipping and 20% temporal rescaling during training. During inference, a 224×224 center crop is simply adopted. Following VAC [15], we employ the VE loss and VA loss for visual supervision, with weights 1.0 and 25.0, respectively. We adopt the TLP loss [40] to extract more powerful representations. Our model is trained and evaluated upon a 3090 graphical card. For the SLT task, the translation network is instantiated as a mBART model [65]. In practice, we found that the gloss labels are beneficial for SLT. Thus we let the translation process additionally supervised with the recognition loss $\mathcal{L}_{\mathrm{CTC}}$, whose final losses can be expressed as: $\mathcal{L}_{\mathrm{T}} = \mathcal{L}_{\mathrm{CTC}} + \mathcal{L}_{\mathrm{CE}}$. We set the learning rate of the visual mapper and translation network as 0.0002 and 1e-6, respectively. We train our models for 40 epochs with learning rates divided by 5 at epoch 20 and 30.

3) Evaluation Metric.: For the **CSLR** task, we use Word Error Rate (WER) as the evaluation metric, which is defined as the minimal summation of the **sub**stitution, **ins**ertion, and **del**etion operations to convert the predicted sentence to the reference sentence, as:

$$\mathrm{WER} = \frac{\#\mathrm{sub} + \#\mathrm{ins} + \#\mathrm{del}}{\#\mathrm{reference}}. \tag{15}$$

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

For the **SLT** task, following previous studies [22], [29], we use commonly-used metrics in machine translation, including tokenized BLEU [66] with ngrams from 1 to 4 (BLEU@1-BLEU@4) and Rouge-L F1 (Rouge) [67] to evaluate the performance of SLT. The higher value, the better performance.

B. Ablation Study

We report ablative results on both development (Dev) and testing (Test) sets of PHOENIX2014 dataset to test the effectiveness of each component in our CorrNet+.

Effectiveness of the proposed modules. Tab. I provides a comprehensive analysis of the effectiveness of the proposed modules. We notice that using any of the proposed three modules yields a notable accuracy boost, with 19.2% & 19.7%, 19.5% & 20.1% accuracy and 19.6 & 20.2% WER on the Dev and Test Sets, respectively. Notably, the correlation module offers the most substantial accuracy improvement. Combining any two modules further activates the effectiveness with 18.4% & 18.7%, 18.8% & 19.2% and 19.0% & 19.3% WER on the Dev and Test Sets, respectively. We notice that combining the correlation module and the identification module gives the most performance promotion. When employing all proposed modules, the accuracy reaches the peak with absolute 18.0% & 18.2% WER, giving +2.2% & +2.8% accuracy boost.

Effects of locations for CorrNet+. Tab II ablates the locations of our proposed modules in Stage 2, 3 or 4. We observe that choosing any one of these locations brings a notable accuracy boost, with 19.3% & 19.9%, 19.2% & 19.7% and 19.0% & 19.5% WER. When combining two or more locations, a larger accuracy gain is witnessed. The accuracy reaches the peak when proposed modules are placed after Stage 2, 3 and 4, with 18.0% & 18.2% accuracy, which is adopted by default.

Study of the effectiveness of correlation module. In the upper part of Tab. III, we first verify the effectiveness of CorrNet+ by comparing it to the CorrNet [30]. By computing correlation maps between all spatial patches among consecutive frames, CorrNet promotes the WER to 18.8% & 19.4% on the Dev and Test sets, respectively. However, it raises substantial computational overhead (3.60 GFLOPs), nearly equivalent to the entire model's computation (3.64 GFLOPs). Instead, by compressing the features of each frame, CorrNet+ notably decreases the incurred computations from

TABLE IV
ABLATIONS FOR THE EFFECTIVENESS OF THE AGGREGATION FUNCTIONS IN CORRELATION MODULE ON THE PHOENIX2014 DATASET.

Aggregation function	Dev(%)	Test(%)
AvgPool	18.5	18.8
AvgPool & MaxPool	18.3	18.5
AvgPool & MaxPool & AttPool	**18.0**	**18.2**

TABLE V
ABLATIONS FOR THE MULTI-SCALE ARCHITECTURE OF IDENTIFICATION MODULE ON THE PHOENIX14 DATASET.

Configuration	Dev(%)	Test(%)
-	20.2	21.0
N_t=4, N_s=**1**	18.8	18.9
N_t=4, N_s=**2**	18.4	18.6
N_t=4, N_s=**3**	**18.0**	**18.2**
N_t=4, N_s=**4**	18.3	18.5
N_t=**2**, N_s=3	18.6	18.7
N_t=**3**, N_s=3	18.3	18.5
N_t=**4**, N_s=3	**18.0**	**18.2**
N_t=**5**, N_s=3	18.5	18.6
K_t=**9**, K_s=**7**	19.1	19.2

3.60 GFLOPs to 0.01 GFLOPs and brings +0.8% & +1.2% accuracy boost, achieving a better accuracy-computation trade-off. In the lower part of Tab. III, we investigate the effects of the temporal receptive filed $L = \{L_1, L_2, L_3\}$ across three network stages for the correlation module. When disabling L, the model degenerates into our baseline. We observe that when setting $L = [2, 2, 2]$ (focusing solely on adjacent frames) CorrNet+ outperforms the baseline by 1.2% & 2.0% on the Dev and Test sets, respectively. Gradually increasing L from [1,1,1] to [5,5,5] consistently improves accuracy with similar computational costs. We then investigate different configurations for the temporal receptive fields as network stages progress. We notice that $L = [2, 6, 10]$ yields the peak accuracy, and either reversing the order of L of further increasing L would degrade the performance.

Study on the effectiveness of aggregation functions in correlation module. We verify the effectiveness of the aggregation functions for the correlation module in Tab. IV. It's observed that by solely using the average aggregation function, CorrNet+ already achieve better results (18.5% & 18.8%) than CorrNet (18.8% & 19.2%). When incorporating both the maximum and attention aggregation functions, the performance is further promoted to 18.3% & 18.5% and 18.0% & 18.2%, underscoring the complementarity of the proposed aggregation functions.

Study on the multi-scale architecture of identification module. In Tab. V, without identification module, our baseline achieves 20.2% and 21.0% WER on the Dev and Test Set, respectively. The base kernel size is set as $3 \times 3 \times 3$ for $K_t \times K_s \times K_s$. When fixing N_t=4 and varying spatial dilation rates to expand spatial receptive fields, a larger N_s consistently brings better accuracy. When N_s reaches 3, it brings no more accuracy gain. Consequently, we set N_s as 3 by default and investigate the impact of N_t. Notably, increasing N_t to 5 or decreasing N_t to 2 and 3 achieves worse accuracy. We thus adopt N_t as 4 by default. We also compare our proposed multi-scale architecture with a normal implementation of more parameters. The receptive field of the identification module with N_t=4, N_s=3 is identical to a normal convolution with K_t=9 and K_s=7. As shown in the bottom of Tab. V, although a normal convolution owns more parameters and computations than ours, it performs worse than our method which verifies the effectiveness of our proposed architecture.

TABLE VI
ABLATIONS FOR THE CONFIGURATIONS OF TEMPORAL ATTENTION MODULE ON THE PHOENIX2014 DATASET.

Configuration	Dev(%)	Test(%)
-	20.2	21.0
M_t=1	18.6	18.7
M_t=2	18.3	18.5
M_t=3	**18.0**	**18.2**
M_t=4	18.2	18.4
M_t=5	18.3	18.5
P_t=5	19.1	19.2
$\boldsymbol{U} \odot \boldsymbol{y}$	21.2	22.1
$\boldsymbol{U} \odot \boldsymbol{y} + \boldsymbol{y}$	19.6	20.3
$(\boldsymbol{U} - 0.5) \odot \boldsymbol{y}$	18.5	18.8
$(\boldsymbol{U} - 0.5) \odot \boldsymbol{y} + \boldsymbol{y}$	**18.0**	**18.2**

TABLE VII
ABLATIONS FOR THE GENERALIZABILITY OF CORRNET OVER MULTIPLE BACKBONES ON THE PHOENIX2014 DATASET.

Configuration	Dev(%)	Test(%)
SqueezeNet [56]	22.2	22.6
w/ CorrNet+	**19.4**	**19.6**
ShuffleNet V2 [68]	21.7	22.2
w/ CorrNet+	**19.1**	**19.5**
GoogleNet [69]	21.4	21.5
w/ CorrNet+	**18.9**	**19.0**
RegNetX-800mf [70]	20.4	21.2
w/ CorrNet+	**18.3**	**18.4**
RegNetY-800mf [70]	20.1	20.8
w/ CorrNet+	**17.8**	**18.0**

Study on the configurations of temporal attention module. In the upper part of Tab. VI, we investigate the effects for the number of branches M_t in the temporal attention module. We notice that as M_t increases, the performance consistently rises until it reaches 3, and a larger M_t can't bring more performance gain. We thus set $M_t = 3$ by default. We then investigate the efficacy of the multiscale architecture by comparing it against a large convolution with kernel size P_t of 5, which has the same temporal receptive field. We observe that our design outperforms it by a large margin with lower computational costs. In the lower part of Tab. VI, we explore the implementations of the temporal attention module to augment original features. Initially, a direct multiplication of the attention maps $\boldsymbol{U}$ with input features y severely degrades performance due to the disruption of input feature distributions. However, when implemented residually by adding y, the expression $\boldsymbol{U} \odot \boldsymbol{y} + \boldsymbol{y}$ notably mitigates this phenomenon, resulting in performance gains of +0.6% and +0.7% on the Dev and Test Sets, respectively. We further subtract 0.5 from the attention maps $\boldsymbol{U}$ to emphasize or

TABLE VIII
COMPARISON WITH OTHER METHODS OF SPATIAL-TEMPORAL ATTENTION OR TEMPORAL REASONING ON THE PHOENIX2014 DATASET.

Method	Dev(%)	Test(%)
-	20.2	21.0
w/ SENet [56]	19.8	20.4
w/ CBAM [71]	19.7	20.2
w/ NLNet [72]	-	-
I3D [5]	22.6	22.9
R(2+1)D [73]	22.4	22.3
TSM [23]	19.9	20.5
CorrNet+	**18.0**	**18.2**

TABLE IX
COMPARISON WITH OTHER METHODS THAT EXPLICITLY EXPLOIT HAND AND FACE FEATURES ON THE PHOENIX2014 DATASET.

Method	Dev(%)	Test(%)
CNN+HMM+LSTM [26]	26.0	26.0
DNF [13]	23.1	22.9
STMC [25]	21.1	20.7
C^2SLR [16]	20.5	20.4
CorrNet+	**18.0**	**18.2**

suppress certain positions, and then element-wisely multiply it with $\boldsymbol{y}$. This refined implementation brings +1.1% & +1.5% performance boost. Finally, we update this implementation in a residual way by adding input features $\boldsymbol{y}$ as $(\boldsymbol{U}-0.5)\odot\boldsymbol{y}+\boldsymbol{y}$, achieving a notable performance boost by +2.2% & +2.8%.

Generalizability of CorrNet+. We deploy CorrNet+ upon multiple backbones, including SqueezeNet [56], ShuffleNet V2 [68], GoogLeNet [69], RegNetX-800mf [70] and RegNetY-800mf [70] to validate its generalizability in Tab. VII. It's observed that our proposed model generalizes well upon different backbones, bringing +2.8% & +3.0%, +2.6% & +2.7%, +2.5% & +2.5%, +2.1% & +2.8% and +2.3% & +2.8% accuracy boost on the Dev and Test Sets, respectively.

Comparisons with other spatial-temporal reasoning methods. Tab. VIII compares our approach with other methods of spatial-temporal reasoning ability. SENet [56] and CBAM [71] perform channel attention to emphasize key information. NLNet [72] employs non-local means to aggregate spatial-temporal information from other frames. I3D [5] and R(2+1)D [73] deploys 3D or 2D+1D convolutions to capture spatial-temporal features. TSM [23] adopts temporal shift operation to obtain features from adjacent frames. In the upper part of Tab. VIII, one can see CorrNet+ largely outperforms other attention-based methods, i.e., SENet, CBAM and NLNet, for its superior ability to identify and aggregate body trajectories. NLNet is out of memory due to its quadratic computational complexity with spatial-temporal size. In the bottom part of Tab. VIII, we observed that I3D and R(2+1)D demonstrate degraded accuracy, which may be attributed to their limited spatial-temporal receptive fields and increased training complexity. TSM slightly brings 0.3% & 0.3% accuracy boost. Our proposed approach significantly outperforms these methods, affirming its efficacy in aggregating salient spatial-temporal information from even distant spatial neighbors.

TABLE X
COMPARISON WITH STATE-OF-THE-ART METHODS ON THE PHOENIX2014 AND PHOENIX2014-T DATASETS OVER THE CSLR SETTING. $*$ INDICATES EXTRA CLUES SUCH AS FACE OR HAND FEATURES ARE INCLUDED BY ADDITIONAL NETWORKS OR PRE-EXTRACTED HEATMAPS.

Method	PHOENIX2014				PHOENIX2014-T	
	Dev(%)		Test(%)		Dev(%)	Test(%)
	del/ins	WER	del/ins	WER		
SFL [14]	7.9/6.5	26.2	7.5/6.3	26.8	25.1	26.1
FCN [12]	-	23.7	-	23.9	23.3	25.1
CMA [38]	7.3/2.7	21.3	7.3/2.4	21.9	-	-
VAC [15]	7.9/2.5	21.2	8.4/2.6	22.3	-	-
SMKD [17]	6.8/2.5	20.8	6.3/2.3	21.0	20.8	22.4
CVT-SLR [21]	6.4/2.6	19.8	6.1/2.3	20.1	19.4	20.3
TLP [40]	6.3/2.8	19.7	6.1/2.9	20.8	19.4	21.2
CoSign-2s [74]	-	19.7	-	20.1	19.5	20.1
AdaSize [75]	7.0/2.6	19.7	7.2/3.1	20.9	19.7	21.2
AdaBrowse+ [76]	6.0/2.5	19.6	5.9/2.6	20.7	19.5	20.6
SEN [18]	5.8/2.6	19.5	7.3/4.0	21.0	19.3	20.7
CTCA [20]	6.2/2.9	19.5	6.1/2.6	20.1	19.3	20.3
RadialCTC [39]	6.5/2.7	19.4	6.1/2.6	20.2	-	-
SLT* [28]	-	-	-	-	24.5	24.6
C+L+H* [26]	-	26.0	-	26.0	22.1	24.1
DNF* [13]	7.3/3.3	23.1	6.7/3.3	22.9	-	-
STMC* [25]	7.7/3.4	21.1	7.4/2.6	20.7	19.6	21.0
C^2SLR* [16]	-	20.5	-	20.4	20.2	20.4
CorrNet+	5.3/2.7	**18.0**	5.6/2.4	**18.2**	**17.2**	**19.1**

TABLE XI
COMPARISON WITH STATE-OF-THE-ART METHODS ON THE CSL-DAILY DATASET [29] OVER THE CSLR SETTING.

Method	Dev(%)	Test(%)
BN-TIN [29]	33.6	33.1
FCN [12]	33.2	32.5
Joint-SLRT [42]	33.1	32.0
TIN-Iterative [13]	32.8	32.4
CTCA [20]	31.3	29.4
AdaSize [75]	31.3	30.9
AdaBrowse+ [76]	31.2	30.7
SEN [18]	31.1	30.7
CorrNet+	**28.6**	**28.2**

TABLE XII
COMPARISON WITH STATE-OF-THE-ART METHODS ON THE CSL DATASET [61] OVER THE CSLR SETTING.

Method	WER(%)
LS-HAN [61]	17.3
SubUNet [77]	11.0
SF-Net [78]	3.8
FCN [12]	3.0
STMC [25]	2.1
VAC [15]	1.6
C^2SLR [16]	0.9
SEN [18]	0.8
CorrNet+	**0.7**

Comparisons with previous methods equipped with hand or face features. Many previous CSLR methods explicitly leverage hand and face features for better recognition by employing multiple input streams [26], human body keypoints [16], [25] and pre-extracted hand patches [13]. They require extra resource-intensive pose-estimation networks like HRNet [79] or additional multiple training stages. Our approach doesn't rely on extra supervision and could be end-to-end trained to dynamically attend to body trajectories like hand and face actions in a self-motivated way. Tab. IX shows that our method could outperform these methods by a large margin with much fewer computations.

C. Comparison with State-of-the-Art Methods

We verify the effectiveness of our proposed method upon two sign language understanding tasks, i.e., continuous sign language recognition (CSLR) and sign language translation (SLT). We next introduce the results of our method upon both settings, respectively.

TABLE XIII
COMPARISON WITH STATE-OF-THE-ART METHODS ON THE PHOENIX2014-T DATASET [28] AND CSL-DAILY DATASET [29] OVER THE SLT SETTING.

PHOENIX2014-T											
	Method	Dev					Test				
		Rouge	BLEU1	BLEU2	BLEU3	BLEU4	Rouge	BLEU1	BLEU2	BLEU3	BLEU4
Sign2Gloss2Text	SL-Luong [28]	44.14	42.88	30.30	23.02	18.40	43.80	43.29	30.39	22.82	18.13
	SignBT [29]	49.53	49.33	36.43	28.66	23.51	49.35	48.55	36.13	28.47	23.51
	STMC-Transf [80]	46.31	48.27	35.20	27.47	22.47	46.77	48.73	36.53	29.03	24.00
	MMTLB [43]	50.23	50.36	37.50	29.69	24.63	49.59	49.94	37.28	29.67	24.60
	TwoStream-SLT [22]	52.01	52.35	39.76	31.85	26.47	51.59	52.11	39.81	32.00	26.71
	SLTUNET [81]	49.61	-	-	-	25.36	49.98	50.42	39.24	31.41	26.00
Sign2Text	SL-Luong [28]	31.80	31.87	19.11	13.16	9.94	31.80	32.24	19.03	12.83	9.58
	Joint-SLRT [42]	-	47.26	34.40	27.05	22.38	-	46.61	33.73	26.19	21.32
	STMC-T [82]	48.24	47.60	36.43	29.18	24.09	46.65	46.98	36.09	28.70	23.65
	SignBT [29]	50.29	51.11	37.90	29.80	24.45	49.54	50.80	37.75	29.72	24.32
	MMTLB [43]	53.10	53.95	41.12	33.14	27.61	52.65	53.97	41.75	33.84	28.39
	SLTUNET [81]	52.23	-	-	-	27.87	52.11	52.92	41.76	33.99	28.47
	TwoStream-SLT [22]	54.08	54.32	41.99	34.15	28.66	53.48	54.90	42.43	34.46	28.95
	CorrNet+	**54.54**	**54.56**	**42.31**	**34.48**	**29.13**	**53.76**	**55.32**	**42.74**	**34.86**	**29.42**
CSL-Daily											
	Method	Dev					Test				
		Rouge	BLEU1	BLEU2	BLEU3	BLEU4	Rouge	BLEU1	BLEU2	BLEU3	BLEU4
Sign2Gloss2Text	SL-Luong [28]	40.18	41.46	25.71	16.57	11.06	40.05	41.55	2573	16.54	11.03
	SignBT [29]	48.38	50.97	36.16	26.26	19.53	48.21	50.68	36.00	26.20	19.67
	MMTLB [43]	51.35	50.89	37.96	28.53	21.88	51.43	50.33	37.44	28.08	21.46
	SLTUNET [81]	52.89	-	-	-	22.95	53.10	54.39	40.28	30.52	23.76
	TwoStream-SLT [22]	53.91	53.58	40.49	30.67	23.71	54.92	54.08	41.02	31.18	24.13
Sign2Text	SL-Luong [28]	34.28	34.22	19.72	12.24	7.96	34.54	34.16	19.57	11.84	7.56
	SignBT [29]	49.49	51.46	37.23	27.51	20.80	49.31	51.42	37.26	27.76	21.34
	MMTLB [43]	53.38	53.81	40.84	31.29	24.42	53.25	53.31	40.41	30.87	23.92
	SLTUNET [81]	53.58	-	-	-	23.99	54.08	54.98	41.44	31.84	25.01
	TwoStream-SLT [22]	55.10	55.21	42.31	32.71	25.76	55.72	55.44	42.59	32.87	25.79
	CorrNet+	**55.52**	**55.64**	**42.78**	**33.13**	**26.14**	**55.84**	**55.82**	**42.96**	**33.26**	**26.14**

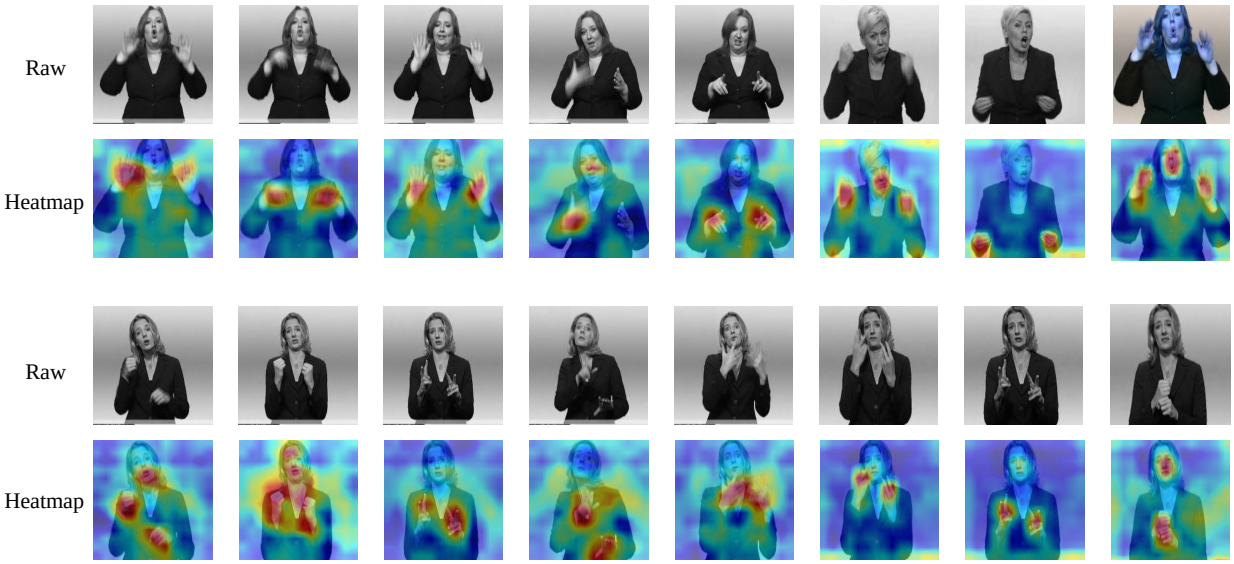


Fig. 7. Visualizations of heatmaps by Grad-CAM [11]. Top: raw frames; Bottom: heatmaps of identification module. Our identification module could generally focus on the human body (light yellow areas) and especially pays attention to informative regions like hands and face (dark red areas) to track body trajectories.

1) Continuous sign language recognition: **PHOENIX2014** and **PHOENIX2014-T**. Tab. VII shows a comprehensive comparison between our CorrNet+ and other state-of-the-art methods. The entries notated with $*$ indicate these methods utilize additional factors like face or hand features for better accuracy. We notice that CorrNet+ outperforms other state-of-the-art methods by a large margin upon both datasets, thanks to its special attention on body trajectories. Especially, CorrNet+ outperforms previous CSLR methods [13], [16], [25], [26] equipped with hand and faces acquired by heavy pose-estimation networks or pre-extracted heatmaps (notated with *), without additional expensive supervision.

CSL-Daily. CSL-Daily is a recently released large-scale dataset with the largest vocabulary size (2k) among commonly-used CSLR datasets, with a wide content covering family life, social contact and so on. Tab. VIII shows that

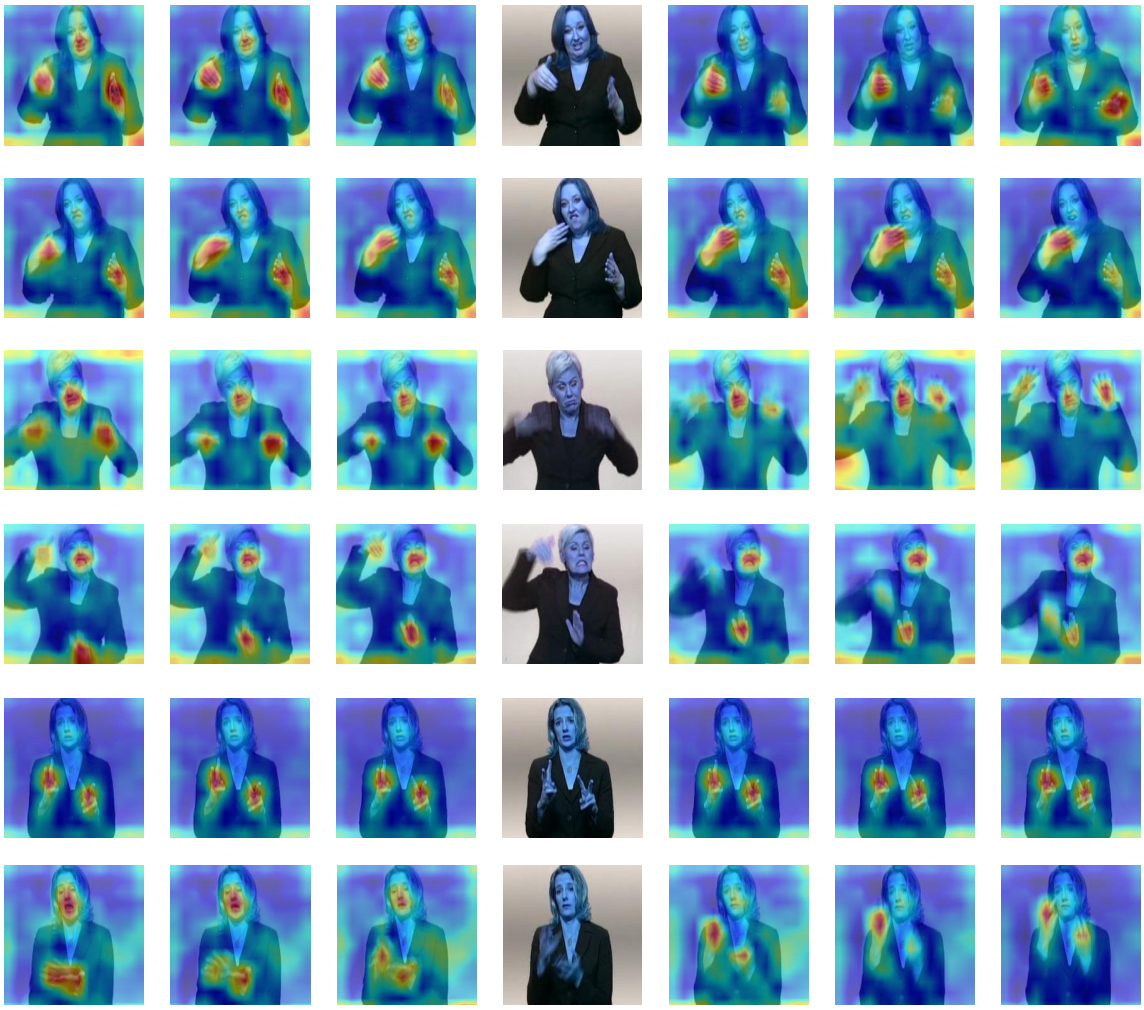


Fig. 8. Visualizations of correlation maps for correlation module. Based on correlation operators, each frame could especially attend to informative regions in adjacent left/right frames like hands and face (dark red areas).

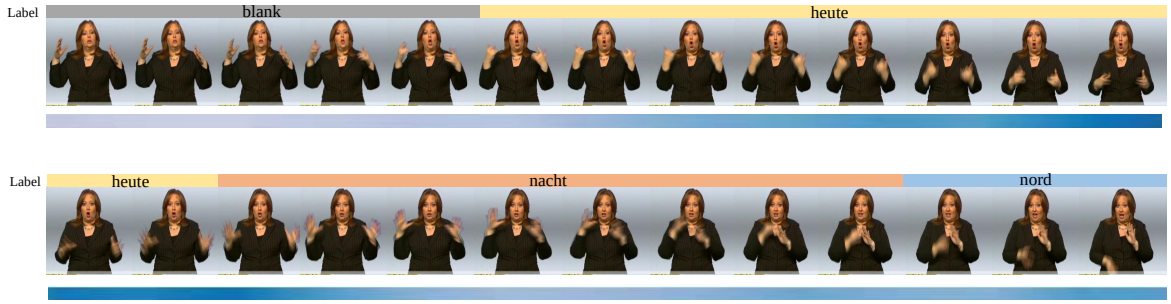


Fig. 9. Visualizations of temporal attention maps for temporal attention module. It's observed that it tends to emphasize frames with rapid movements and suppress those frames with static contents.

our CorrNet+ achieves new state-of-the-art accuracy upon this challenging dataset with notable progress, which generalizes well upon real-world scenarios.

CSL. As shown in Tab. IX, our CorrNet+ could achieve extremely superior accuracy (0.7% WER) upon this well-examined dataset, outperforming existing CSLR methods.

2) Sign language translation: We compare our method with recent methods upon two widely-used SLT datasets, **Phoenix-2014T** and **CSL-Daily**, in Tab. XIII. These methods are roughly divided into two categories, *Sign2Gloss2Text* which first transforms input videos into intermediate gloss representations and then performs translation, and *Sign2Text* which directly conducts end-to-end translation from input videos. We observe that our method outperforms previous methods across both datasets, demonstrating its effectiveness in sign language comprehension. Especially, the powerful TwoStream-SLT [22] adopts both RGB videos and skeleton data as inputs to fuse beneficial information from both modalities,

which requires more expensive supervision and heavy computations. In contrast, our method achieves better performance by only inputting RGB videos, demonstrating a better accuracy-computation trade-off.

D. Visualizations

Visualizations for identification module. Fig. 7 shows the heatmaps generated by our identification module. Our identification module pays special attention to the human body (light yellow areas), especially informative regions of hands and face (dark red areas) to capture human body trajectories. These results verify the effectiveness of our identification module in dynamically emphasizing critical areas in expressing sign language and suppressing other background regions to overlook noisy information.

Visualizations for correlation module. Fig. 8 illustrates the correlation maps generated by our correlation module, which shows the computed spatial-temporal correlations between the current frame and temporal neighboring frames. Three adjacent frames are shown to visualize the correlation maps. We observe that our correlation module pays major attention to informative regions in adjacent frames like hands or the face to enable precise tracking of body trajectories during sign expression. Especially, it learns to focus on the moving body parts that play a major role in expressing signs to enhance sign language comprehension. For example, in the 3rd and 4th row, the correlation module consistently pays major attention to the quickly moving right hands to capture sign information while overlooking the redundant information in the background.

Visualizations for temporal attention module. Fig. 9 visualizes the temporal attention maps generated by our temporal attention module over some selected frames. The darker color, the higher value. We observe that our temporal attention module tends to allocate higher weights for frames with rapid movements (e.g., the latter several frames in the first line; the frontal frames in the second line). It learns to assign lower weights for static frames with few body movements. This observation is consistent with the habits of our human beings, as our humans always pay more attention to those moving objects in the visual field to capture key movements. These observations clearly reveal the effectiveness of our temporal attention module in emphasizing the critical segments in the whole sign video.

V. Conclusion

Recent methods on sign language understanding usually solely focus on each frame to extract their spatial features and overlook their cross-frame interactions, thus failing to capture the key human body movements. To handle this problem, this paper introduces an enhanced correlation network (CorrNet+) to capture human body trajectories, which comprises a correlation module, an identification module and a temporal attention module. The effectiveness of CorrNet+ is verified on two sign language understanding tasks including continuous sign language recognition (CSLR) and sign language translation (SLT) with new state-of-the-art performance compared to previous methods. Especially, by only inputting RGB videos on both tasks, CorrNet+ outperforms previous methods equipped with resource-intensive pose estimation networks or pre-extracted heatmaps with much fewer computations for hand and facial feature extraction. Compared to CorrNet [30], CorrNet+ achieves a significant performance boost across multiple benchmarks with drastically reduced computational costs, demonstrating a better accuracy-computation trade-off. Plentiful visualizations further verify the effectiveness of CorrNet+ in intelligently emphasizing human body trajectories across adjacent frames in a self-motivated way.

References

[1] P. Dreuw, D. Rybach, T. Deselaers, M. Zahedi, and H. Ney, "Speech recognition techniques for a sign language recognition system," *hand*, vol. 60, p. 80, 2007.

[2] S. C. Ong and S. Ranganath, "Automatic sign language analysis: A survey and the future beyond lexical meaning," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 27, no. 06, pp. 873–891, 2005.

[3] N. Adaloglou, T. Chatzis, I. Papastratis, A. Stergioulas, G. T. Papadopoulos, V. Zacharopoulou, G. J. Xydopoulos, K. Atzakas, D. Papazachariou, and P. Daras, "A comprehensive study on deep learning-based methods for sign language recognition," *IEEE Transactions on Multimedia*, vol. 24, pp. 1750–1762, 2021.

[4] R. Rastgoo, K. Kiani, and S. Escalera, "Sign language recognition: A deep survey," *Expert Systems with Applications*, vol. 164, p. 113794, 2021.

[5] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[6] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks for action recognition in videos," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 11, pp. 2740–2755, 2018.

[7] Z. Shou, D. Wang, and S.-F. Chang, "Temporal action localization in untrimmed videos via multi-stage cnns," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1049–1058.

[8] P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "Learning to track for spatio-temporal action localization," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3164–3172.

[9] L. Zhu, Z. Xu, Y. Yang, and A. G. Hauptmann, "Uncovering the temporal context for video question answering," *International Journal of Computer Vision*, vol. 124, pp. 409–421, 2017.

[10] Y. Li, X. Wang, J. Xiao, W. Ji, and T.-S. Chua, "Invariant grounding for video question answering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2928–2937.

[11] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[12] K. L. Cheng, Z. Yang, Q. Chen, and Y.-W. Tai, "Fully convolutional networks for continuous sign language recognition," in *ECCV*, 2020.

[13] R. Cui, H. Liu, and C. Zhang, "A deep neural framework for continuous sign language recognition by iterative training," *TMM*, vol. 21, no. 7, pp. 1880–1891, 2019.

[14] Z. Niu and B. Mak, "Stochastic fine-grained labeling of multi-state sign glosses for continuous sign language recognition," in *ECCV*, 2020.

[15] Y. Min, A. Hao, X. Chai, and X. Chen, "Visual alignment constraint for continuous sign language recognition," in *ICCV*, 2021.

[16] R. Zuo and B. Mak, "C2slr: Consistency-enhanced continuous sign language recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5131–5140.

[17] A. Hao, Y. Min, and X. Chen, "Self-mutual distillation learning for continuous sign language recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 303–11 312.

[18] L. Hu, L. Gao, Z. Liu, and W. Feng, "Self-emphasizing network for continuous sign language recognition," in *Thirty-seventh AAAI conference on artificial intelligence*, 2023.

[19] B. Zhou, Z. Chen, A. Clapés, J. Wan, Y. Liang, S. Escalera, Z. Lei, and D. Zhang, "Gloss-free sign language translation: Improving from visual-language pretraining," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 20 871–20 881.

[20] L. Guo, W. Xue, Q. Guo, B. Liu, K. Zhang, T. Yuan, and S. Chen, "Distilling cross-temporal contexts for continuous sign language recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 10 771–10 780.

[21] J. Zheng, Y. Wang, C. Tan, S. Li, G. Wang, J. Xia, Y. Chen, and S. Z. Li, "Cvt-slr: Contrastive visual-textual transformation for sign language recognition with variational alignment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 23 141–23 150.

[22] Y. Chen, R. Zuo, F. Wei, Y. Wu, S. Liu, and B. Mak, "Two-stream network for sign language recognition and translation," *Advances in Neural Information Processing Systems*, vol. 35, pp. 17 043–17 056, 2022.

[23] J. Lin, C. Gan, and S. Han, "Tsm: Temporal shift module for efficient video understanding," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7083–7093.

[24] Z. Liu, D. Luo, Y. Wang, L. Wang, Y. Tai, C. Wang, J. Li, F. Huang, and T. Lu, "Teinet: Towards an efficient architecture for video recognition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 11 669–11 676.

[25] H. Zhou, W. Zhou, Y. Zhou, and H. Li, "Spatial-temporal multi-cue network for continuous sign language recognition," in *AAAI*, 2020.

[26] O. Koller, N. C. Camgoz, H. Ney, and R. Bowden, "Weakly supervised learning with multi-stream cnn-lstm-hmms to discover sequential parallelism in sign language videos," *PAMI*, vol. 42, no. 9, pp. 2306–2320, 2019.

[27] O. Koller, J. Forster, and H. Ney, "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers," *Computer Vision and Image Understanding*, vol. 141, pp. 108–125, 2015.

[28] N. C. Camgoz, S. Hadfield, O. Koller, H. Ney, and R. Bowden, "Neural sign language translation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7784–7793.

[29] H. Zhou, W. Zhou, W. Qi, J. Pu, and H. Li, "Improving sign language translation with monolingual data by sign back-translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1316–1325.

[30] L. Hu, L. Gao, Z. Liu, and W. Feng, "Continuous sign language recognition with correlation network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 2529–2539.

[31] W. Gao, G. Fang, D. Zhao, and Y. Chen, "A chinese sign language recognition system based on sofm/srn/hmm," *Pattern Recognition*, vol. 37, no. 12, pp. 2389–2402, 2004.

[32] W. T. Freeman and M. Roth, "Orientation histograms for hand gesture recognition," in *International workshop on automatic face and gesture recognition*, vol. 12. Zurich, Switzerland, 1995, pp. 296–301.

[33] O. Koller, O. Zargaran, H. Ney, and R. Bowden, "Deep sign: Hybrid cnn-hmm for continuous sign language recognition," in *Proceedings of the British Machine Vision Conference 2016*, 2016.

[34] J. Han, G. Awad, and A. Sutherland, "Modelling and segmenting subunits for sign language recognition based on hand motion analysis," *Pattern Recognition Letters*, vol. 30, no. 6, pp. 623–633, 2009.

[35] O. Koller, S. Zargaran, and H. Ney, "Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms," in *CVPR*, 2017.

[36] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[37] J. Pu, W. Zhou, and H. Li, "Iterative alignment network for continuous sign language recognition," in *CVPR*, 2019.

[38] J. Pu, W. Zhou, H. Hu, and H. Li, "Boosting continuous sign language recognition via cross modality augmentation," in *ACM MM*, 2020.

[39] Y. Min, P. Jiao, Y. Li, X. Wang, L. Lei, X. Chai, and X. Chen, "Deep radial embedding for visual sequence learning," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VI*. Springer, 2022, pp. 240–256.

[40] L. Hu, L. Gao, Z. Liu, and W. Feng, "Temporal lift pooling for continuous sign language recognition," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXV*. Springer, 2022, pp. 511–527.

[41] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[42] N. C. Camgoz, O. Koller, S. Hadfield, and R. Bowden, "Sign language transformers: Joint end-to-end sign language recognition and translation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 023–10 033.

[43] Y. Chen, F. Wei, X. Sun, Z. Wu, and S. Lin, "A simple multi-modality transfer learning baseline for sign language translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5120–5130.

[44] J. Ye, W. Jiao, X. Wang, Z. Tu, and H. Xiong, "Cross-modality data augmentation for end-to-end sign language translation," *arXiv preprint arXiv:2305.11096*, 2023.

[45] D. Zhu, V. Czehmann, and E. Avramidis, "Neural machine translation methods for translating text to sign language glosses," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2023, pp. 12 523–12 541.

[46] K. Lin, X. Wang, L. Zhu, K. Sun, B. Zhang, and Y. Yang, "Gloss-free end-to-end sign language translation," *arXiv preprint arXiv:2305.12876*, 2023.

[47] I. Rocco, R. Arandjelovic, and J. Sivic, "Convolutional neural network architecture for geometric matching," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6148–6157.

[48] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Detect to track and track to detect," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 3038–3046.

[49] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "Deepflow: Large displacement optical flow with deep matching," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 1385–1392.

[50] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2758–2766.

[51] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8934–8943.

[52] X. Shi, Z. Huang, W. Bian, D. Li, M. Zhang, K. C. Cheung, S. See, H. Qin, J. Dai, and H. Li, "Videoflow: Exploiting temporal cues for multi-frame optical flow estimation," *arXiv preprint arXiv:2303.08340*, 2023.

[53] X. Shi, Z. Huang, D. Li, M. Zhang, K. C. Cheung, S. See, H. Qin, J. Dai, and H. Li, "Flowformer++: Masked cost volume autoencoding for pretraining optical flow estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1599–1610.

[54] Y. Zhao, Y. Xiong, and D. Lin, "Recognize actions by disentangling components of dynamics," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6566–6575.

[55] A. Diba, M. Fayyaz, V. Sharma, M. M. Arzani, R. Yousefzadeh, J. Gall, and L. Van Gool, "Spatio-temporal channel correlation networks for action classification," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 284–299.

[56] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[57] M. Lee, S. Lee, S. Son, G. Park, and N. Kwak, "Motion feature network: Fixed motion filter for action recognition," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 387–403.

[58] H. Wang, D. Tran, L. Torresani, and M. Feiszli, "Video modeling with correlation networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 352–361.

[59] Y. Xu, H. Cao, K. Mao, Z. Chen, L. Xie, and J. Yang, "Aligning correlation information for domain adaptation in action recognition," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[60] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[61] J. Huang, W. Zhou, Q. Zhang, H. Li, and W. Li, "Video-based sign language recognition without temporal segmentation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[62] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

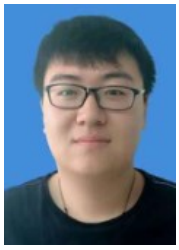
[63] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[64] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[65] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer, "Multilingual denoising pre-training for neural machine translation," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 726–742, 2020.
[66] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.
[67] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.
[68] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 116–131.
[69] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.
[70] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, "Designing network design spaces," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 428–10 436.
[71] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.
[72] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7794–7803.
[73] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 6450–6459.
[74] P. Jiao, Y. Min, Y. Li, X. Wang, L. Lei, and X. Chen, "Cosign: Exploring co-occurrence signals in skeleton-based continuous sign language recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 20 676–20 686.
[75] L. Hu, L. Gao, Z. Liu, and W. Feng, "Scalable frame resolution for efficient continuous sign language recognition," *Pattern Recognition*, vol. 145, p. 109903, 2024.
[76] L. Hu, L. Gao, Z. Liu, C.-M. Pun, and W. Feng, "Adabrowse: Adaptive video browser for efficient continuous sign language recognition," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 709–718.
[77] N. Cihan Camgoz, S. Hadfield, O. Koller, and R. Bowden, "Subunets: End-to-end hand shape and continuous sign language recognition," in *ICCV*, 2017.
[78] Z. Yang, Z. Shi, X. Shen, and Y.-W. Tai, "Sf-net: Structured feature network for continuous sign language recognition," *arXiv preprint arXiv:1908.01341*, 2019.
[79] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 10, pp. 3349–3364, 2020.
[80] K. Yin and J. Read, "Better sign language translation with stmc-transformer," *arXiv preprint arXiv:2004.00588*, 2020.
[81] B. Zhang, M. Müller, and R. Sennrich, "SLTUNET: A simple unified model for sign language translation," in *The Eleventh International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=EBS4C77p_5S
[82] H. Zhou, W. Zhou, Y. Zhou, and H. Li, "Spatial-temporal multi-cue network for sign language recognition and translation," *IEEE Transactions on Multimedia*, vol. 24, pp. 768–779, 2021.

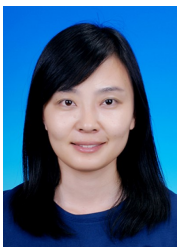
Lianyu Hu received the bachelor degree and master degree in computer science and technology from Dalian University of Technology in 2018 and 2021, respectively. He is currently a Ph.D. candidate with the College of Intelligence and Computing at Tianjin University, China. His research interests include video understanding, multimodal understanding and sign language understanding.

Wei Feng is a full Professor at the School of Computer Science and Technology, College of Computing and Intelligence, Tianjin University, China. He received the PhD degree in computer science from City University of Hong Kong in 2008. His major research interests are active robotic vision and visual intelligence, specifically including active camera relocalization and lighting recurrence, active 3D scene perception, video analysis, and generic pattern recognition.

Liqing Gao received the BS and MS degree in Electronic & Information Engineering, Inner Mongolia University, China, in 2015 and 2018. She is working toward the PhD degree in the College of Intelligence and Computing, Tianjin University, China. Her research interests include sign language recognition and gesture recognition.



Zekang Liu received the BS and ME in Software Engineering from Hebei University of Economics and Business, China and Tianjin Normal University, China, in 2017 and 2019, respectively. He is studying for a Eng.D in the College of Intelligence Computing, Tianjin University, China. His research interests include vehicle detection and sign-language recognition.



Liang Wan is a full Professor in the College of Intelligence Computing, and deputy director of Medical College, Tianjin University, P. R. China. She obtained a Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong in 2007, and worked as a PostDoc Research Associate/Fellow at City University of Hong Kong from 2007 to 2011. Her current research interests focus on image processing and computer vision, including image segmentation, low-level image restoration, and medical image analysis.