Graph-Based Multimodal Sequential Embedding for Sign Language Translation

Shengeng Tang, Dan Guo*, Richang Hong*, and Meng Wang, *IEEE Fellow*

***Abstract*—Sign language translation (SLT) is a challenging weakly supervised task without word-level annotations. An effective method of SLT is to leverage multimodal complementarity and to explore implicit temporal cues. In this work, we propose a graph-based multimodal sequential embedding network (MSeqGraph), in which multiple sequential modalities are densely correlated. Specifically, we build a graph structure to realize the intra-modal and inter-modal correlations. First, we design a graph embedding unit (GEU), which embeds a parallel convolution with channel-wise and temporal-wise learning into the graph convolution to learn the temporal cues in each modal sequence and cross-modal complementarity. Then, a hierarchical GEU stacker with a pooling-based skip connection is proposed. Unlike the state-of-the-art methods, to obtain a compact and informative representation of multimodal sequences, the GEU stacker gradually compresses the channel d with multi-modalities m rather than the temporal dimension t. Finally, we adopt the connectionist temporal decoding strategy to explore the entire video's temporal transition and translate the sentence. Extensive experiments on the USTC-CSL and BOSTON-104 datasets demonstrate the effectiveness of the proposed method.**

***Index Terms*—Continuous sign language translation, graph convolutional network, multimodal sequential embedding, multimodal sequential fusion.**

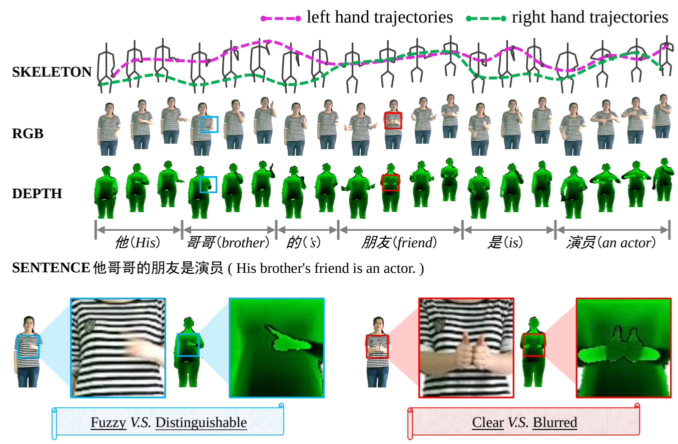


Fig. 1. Continuous SLT with multimodal cues from RGB, depth, and skeleton data. We aim to utilize multimodal cues to correlate and integrate the variations of sign actions. Gesture appearances are performed in RGB images, depth frames distinguish overlapping limbs with depth cues, and skeleton coordinates reflect skeletal joints' trajectories.

I. Introduction

SIGN language bridges the communication gap between deaf-mute and non-disabled people. The goal of sign language translation (SLT) is to convert a video performing continuous signs into a natural language sentence, which is a typical vision-to-text task attracting increasing attention in the research community [1], [2]; it refers to related studies such as video understanding [3], action recognition [4], and video captioning [5]. The current development of the SLT task is limited by some challenges. Complicated and professional sign language linguistics is little known except to linguists. Unlike common video comprehension tasks, subtle but important action variations are difficult to detect in SLT, which are often implied in multi-source sign inputs. As shown in Fig. 1, the multimodal data streams exhibit significant differences along the time dimension. RGB images describe the fingers' details, depth images display the edges of limbs under fast-moving states, and the skeletal coordinates reflect joints' motion trajectories. Leveraging multimodal data can effectively compensate for the deficiencies of each modality. However, it is challenging to bridge the huge semantic gap about data consistency among multimodal inputs. Furthermore, implicit semantic units of signs can be represented at the frame level, clip level, and video level, resulting in difficulty in performing multi-scale temporal cue learning. In addition, weakly supervised sequential learning remains to be solved without exact word annotation.

Early SLT works were dedicated to exploring the spatiotemporal implications in videos. In such early works, frame-level features are extracted and fed to a sequential learning network to model temporal associations [1], [7], [8]. Then, current feature extraction, referring to methods adopting three-dimensional convolutional neural networks (3D CNNs) to learn the spatiotemporal cues simultaneously are used [9], [10]. To refine the feature representation of videos, some multi-stream fusion methods are adopted for sign language interpretation [11]–[13], such as integrating original 3D CNN features and (2D+1D) CNN features [13] and score fusion [11]. To further address the weakly supervised issue in sequential learning, some works have improved the architecture of neural networks, such as HLSTM [14], pyramid BiLSTM [15] and the transformer-based model [2]. Pseudo-supervised optimization based on the expectation-

This work is supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61876058, Grant 61932009, Grant U20A20183, Grant 62020106007, and Fundamental Research Funds for the Central Universities under Grant JZ2020HGTB0020. (Corresponding author: D. Guo and R. Hong.)

S. Tang, D. Guo, R. Hong, and M. Wang are with [1] Key Laboratory of Knowledge Engineering with Big Data (HFUT), Ministry of Education, [2] Intelligent Interconnected Systems Laboratory of Anhui Province (HFUT) and [3] School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230601, China (e-mail: tsg1995@mail.hfut.edu.cn; guodan@hfut.edu.cn; hongrc.hfut@gmail.com; eric.mengwang@gmail.com).

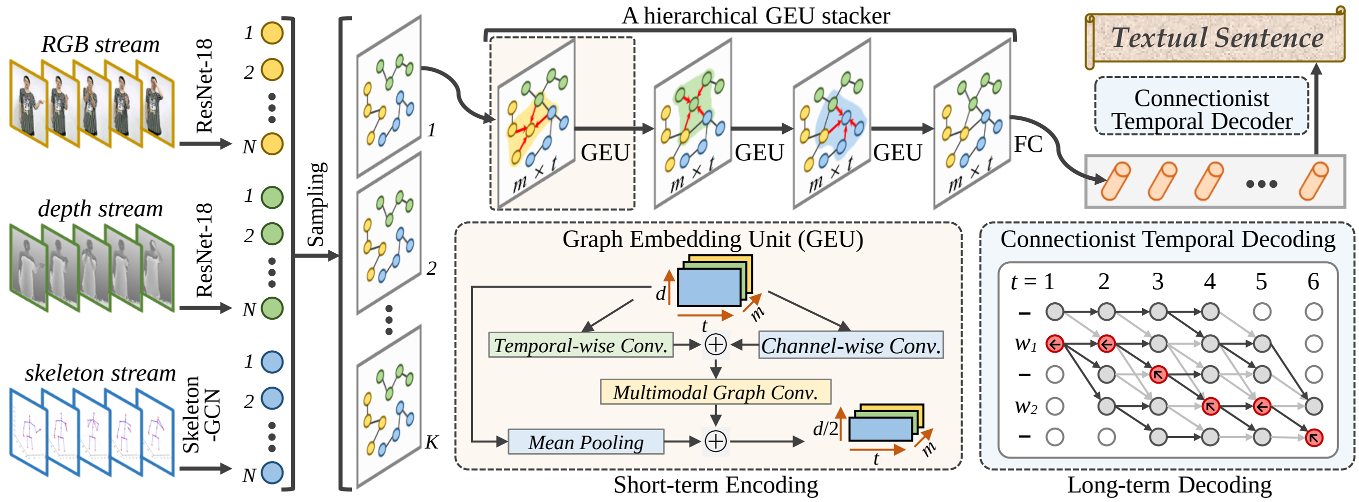


Fig. 2. Overview of the proposed MSeqGraph framework for SLT. Given a sign video, we use the pretrained ResNet-18 [6] and the proposed skeletal-GCN model to extract RGB, depth, and skeleton features. These features are then grouped into K clips by sampling t continuous frames; then, each clip (t continuous frames) with m modalities is fed into the GEU for intra-modal and inter-modal correlations. A hierarchical GEU stacker is applied to deeply exploit the representation learning of video. After that, each feature map $\mathcal{F} \in \mathbb{R}^{m \times t \times d}$ is transformed into a fused vector by an FC layer. Thus, with K feature maps, we obtain a new feature sequence $\mathcal{G} = \{g_k\}_{k=1}^{K}$. Finally, we utilize a connectionist temporal decoder to generate a sentence, which assesses all possible decoding paths along the video's entire temporal dimension.

maximization algorithm is also used in weakly supervised learning for SLT [8], [9], [13], [16], [17]. Existing methods have usually focused on vision-based SLT. Less work has explored skeletal features. The common method of processing of skeleton data is to concatenate 3D coordinates into a vector [18], [19], or to tackle the spatial distribution of joints as an image and extract pseudo 'visual' features [20]. These methods ignore dynamically modeling the spatial correlation among joint points. In contrast, we introduce multi-source cues into SLT and design a Skeleton-GCN network, which builds a skeleton graph to learn the relation among joints.

Exploring implicit modal cues through fusion and interaction has been promising in improving a variety of multimodal and cross-modal tasks [21]–[23]. Classical fusion methods are divided into feature fusion [5], [24], [25] and score fusion [11], [26], [27]. Recently, to enhance the robustness of multimodal representation, feature embedding and the joint aggregation of multi-stream features have been exploited in new tasks such as multimodal understanding tasks [28]–[30] and cross-modal reasoning tasks [31]–[33]. The models in these studies belong to neither feature nor score fusion but a better representation learning of multimodal cues. Using either isolated or continuous multimodal features, these methods calculate the global correlation among multimodal features and usually output an integrated embedding variable to decode, predict, or generate the tasks' answers. In this paper, the SLT task is different. We tackle the multimodal sequential data and output sequential embedding features. Specifically, we learn the multimodal sequential data in a gradually aggregated manner. In addition, the proposed MSeqGraph simultaneously learns inter-modal complementarity and explores intra-modal spatiotemporal cues in the sequential learning process. We hope our work inspires related tasks of multimodal sequential learning.

In this work, we aim to utilize multimodal cues to correlate and integrate the temporal variations of multiple modalities. To this end, we propose a graph-based multimodal sequential embedding network (MSeqGraph) for SLT, as shown in Fig. 2. Given a sign video with multiple modalities, the pretrained ResNet-18 [6] is used to extract RGB and depth features. A joint-based Skeleton-GCN model is proposed for extracting skeleton features. The above features are fed into a graph embedding unit (GEU) for parallel temporal-wise and channel-wise learning and multimodal relational graph embedding. In addition, we further design a hierarchical stacker that concatenates multiple GEUs to capture dense feature embedding. Through the core concept (*i.e.*, GEU) learning the inter-modal complementarity and intra-modal spatiotemporal cues in addition to a common FC layer, we obtain a compact and informative sequential embedding representation. Finally, we utilize the CTC optimizer to decode the feature sequence and translate it into a sentence. The main contributions are summarized as follows:

- We propose a novel graph-based multimodal sequential embedding network, MSeqGraph, which designs a GEU stacker to capture multimodal information and temporal cues, thereby obtaining compact, complementary, and informative representation of the video.
- The Skeleton-GCN model is proposed to learn the spatial characteristics of skeleton joints, where the edge relation (adjacency matrix) in the joint graph is built according to body connectivity.
- The GEU consists of channel-wise embedding, temporal embedding, and multimodal embedding operations guaranteed by the PCN (temporal-wise convolution and channel-wise convolution in parallel) and GCN (multimodal graph relational learning). Unlike state-of-the-art methods compressing the temporal dimension t, we compact the channel d to aggregate multimodal sequential representation. A hierarchical GEU stacker is used to aggregate the densely correlated multimodal representa-

tions.

- Extensive experiments on two benchmark datasets (*i.e.*, USTC-CSL and BOSTON-104) demonstrate the effectiveness of the proposed MSeqGraph. Ablation studies and qualitative visualizations also verify each component of MSeqGraph.

The rest of this paper is organized as follows. Section II reviews the related works. The proposed MSeqGraph model is elaborated in Section III. Implementation details and experimental results are provided in Section IV. Finally, we conclude in Section V.

II. Related Work

This section reviews related work on sign language translation, multimodal fusion, and graph neural networks.

A. Sign Language Translation

The sign language translation (SLT) task [1], [34], [35] was developed from isolated sign language recognition (SLR) [25], [36], which mainly involves feature representation and sequential learning. In early works [37]–[39], hand-crafted features were utilized for identifying different sign actions. With the development of deep learning, various deep representations of sign actions in videos have emerged, such as 2D CNN features [40], 3D CNN features [12], and optical flow [3]. To address the sequential learning issue in SLR and SLT tasks, traditional sequential methods, such as hidden Markov models (HMMs) [7] and dynamic time warping (DTW) [41], are widely used. Considering the merits of CNNs for feature extraction and RNNs for sequential learning, hybrid CNN & RNN models have emerged [8], [11]. The state-of-art works always extracted clip-level features of each video, *i.e.*, compacting $T \rightarrow \frac{T}{16}$, under each modality. In this case, the complementarity of multi-modality along the timeline was ignored. In this paper, we devote ourselves to the complementary and informative representation learning. We embed the frame-level feature and compact the channel dimension instead of the temporal dimension to further learn the fine-grained multimodal temporal cues.

To further address the weakly supervised issue in sequential learning, some works have utilized different architectures of neural networks. For example, Guo *et al.* [14] proposed a hierarchical-RNN network with visual encoding and word embedding, which mainly captured visual cues of different granularities. Li *et al.* [15] constructed a pyramid BiLSTM structure to capture key actions by searching the salient responses. Camgoz *et al.* [42] combined CNNs and an attention-based encoder-decoder to translate sign videos into spoken language, and then they [2] used a transformer-based model rather than a RNN to bind the two sequence-to-sequence issues (*i.e.* recognition and translation) into a unified architecture. In addition, weakly supervised learning in SLT has been researched through pseudo-supervision methods [9], [16]. Specifically, researchers used a multi-stage translation framework to obtain pseudo labels, and fine-tuned the feature extractor, and then alternatively optimized the multi-stage translation module and the feature extraction module [8], [9]. A typical offline optimization method is named expectation-maximization (EM), *e.g.*, the usage of EM in Stage-Opt [16] and CNN-Hybrid [17]. Moreover, in [13], Guo *et al.* proposed an online pseudo-supervised learning solution through an end-to-end connectionist temporal decoding model.

B. Multimodal Embedding & Fusion

Leveraging multimodal cues is quite common in various artificial intelligence tasks, *e.g.*, cross-modal retrieval [21], multimodal action recognition [22], and audio-visual speech enhancement [23]. The classical fusion mechanism is divided into feature fusion and score fusion. Feature fusion is devoted to capturing the correlation among different modalities by concatenation [24] or element-wise summations [43]. Score fusion integrates the score probabilities from different modalities, rather than modeling cross-modal interaction [26], [27]. Wang *et al.* [11] designed a hybrid network containing TCOV, BGRU, and FL modules to capture local, global, and mutual patterns of visual features and performed score fusion. Guo *et al.* [25] proposed an early-late fusion, which first concatenated RGB and depth features into a combined feature and then adaptively selected RGB, depth, and the combined feature. Furthermore, some SLT methods extract multi-channel or multi-cue features from single-source original data for complementary learning. Camgoz *et al.* [44] modeled sign videos by incorporating both manual features and non-manual features, and proposed a multi-channel transformer to capture inter- and intrachannel contextual relationships. Yin *et al.* [45] used a spatial multi-cue module to decompose the input video into spatial features of multiple visual cues and a temporal multi-cue module to calculate temporal correlations at different time steps. To the best of our know, no work has ever focused on multimodal sequential embedding learning in the field of SLT. Existing works have usually addressed multimodal embedding and sequential modeling as two independent parts. The convention is that after feature extraction (independently unimodal), multimodal embedding or fusion is performed at first (jointly multimodal), and sequential learning (sentence generation) is then performed. We propose the GEU module to sequentially learn the multimodal embedding and fusion (jointly multimodal), which focuses on the fine-grained multimodal complementarity along the timeline.

Recently, feature embedding and the joint aggregation of multi-stream features have been exploited to enhance the robustness of multimodal representation in some new tasks, such as multimodal understanding tasks [28]–[30] and cross-modal reasoning tasks [31]–[33]. The models in these studies belong to neither feature nor score fusion but a better representation learning of multimodal cues. Attention-based fusion has become popular. Yu *et al.* [46] embedded multimodal factorized bilinear (MFB) pooling into a novel co-attention mechanism. Tensor fusion network (TFN) [47] explored an outer product correlation between different modalities. Low-rank multimodal fusion (LMF) [48] improved the matrix learning in TFN by using low-rank vector decomposition, thereby reducing the number of parameters. Using either isolated or continuous multimodal features, these methods calculate the global correlation among multimodal features and usually output an integrated embedding variable to decode, predict, or generate the tasks' answers.

In this paper, we emphasize complementary learning along the timeline at the frame level. Therefore, channel embedding, temporal embedding and multimodal embedding are innovatively integrated into the same graph embedding unit, namely the GEU module. The proposed MSeqGraph simultaneously learns inter-modal complementarity and explores intra-modal spatiotemporal cues in the sequential learning process, and outputs a new feature embedding sequence. We hope our method will inspire related works of multimodal sequential learning.

C. Graph Neural Network

Graph neural networks (GNNs) are widely applied to relational learning in various tasks, such as image semantic segmentation [49], neural machine translation [50] and recommendation systems [51]. GNNs have also effectively addressed action recognition. Yan *et al.* [4] constructed the intrabody edges and interframe edges in consecutive multiple skeleton frames and utilized GNN to capture both the spatial and temporal variations of motion in the video. Ye *et al.* [52] proposed a dynamical multi-scale GNN that modeled the relations among body joints for motion-level feature learning. It is reasonable to apply GNN to model the variation of sign actions in the SLT task. The existing GNN-based works in SLT merely model skeletal variations as in the above-mentioned action recognition and ignore the relational learning of multimodalities [53], [54]. A Skeleton-GCN is designed to learn more robust skeleton representation from the joint coordinates in our work. We also leverage the GNN-based model to capture the intra-modal temporal correlations and inter-modal complementarity among three different modalities of data.

III. Proposed Method

As depicted in Fig. 2, the overall pipeline of the proposed approach consists of three steps: feature extraction in Sec. III-A, multimodal sequential embedding in Sec. III-B, and connectionist temporal decoding in Sec. III-C. Given a video containing three multimodal data streams with N frames, we first obtain feature sequences $\mathcal{V} = \{v_n^a|_{n=1}^N, v_n^d|_{n=1}^N, v_n^s|_{n=1}^N\}$ (*i.e.*, RGB feature v_n^a, depth feature v_n^d, and skeleton feature v_n^s), and then propose a graph-based multimodal sequential embedding scheme to aggregate these different multimodal sequential cues into an integrated feature sequence $\mathcal{G} = \{g_k\}_{k=1}^K$. Finally, we decode them into a generated sentence of gloss labels $\mathcal{W} = \{w_l\}_{l=1}^L$.

A. Feature Extraction

For the RGB and depth frames of each video, we use ResNet-18 [6] to obtain RGB features $\mathcal{V}^a = \{v_n^a\}_{n=1}^N$ and depth features $\mathcal{V}^d = \{v_n^d\}_{n=1}^N$. Regarding skeleton data, considering the spatial distribution of 3D coordinates, we propose a graph-based Skeleton-GCN model. The skeleton features $\mathcal{V}^s = \{v_n^s\}_{n=1}^N$ are extracted by Skeleton-GCN. Here,

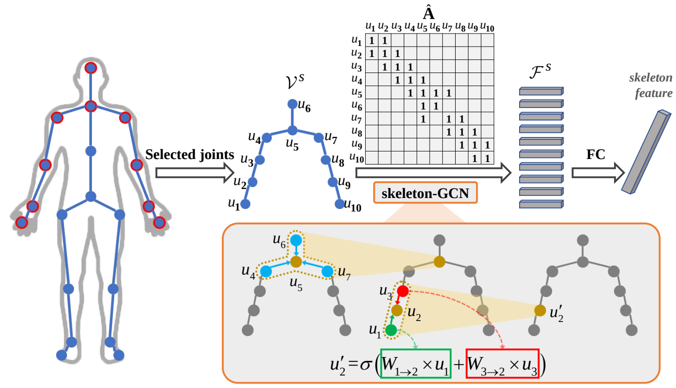


Fig. 3. Graph embedding of skeletal joints in Skeleton-GCN. If u_i and u_j are connected, the element a_{ij} in $\widehat{\mathbf{A}}$ is assigned to 1; otherwise, it is assigned to 0. Each node is updated by messages from its neighboring nodes. Taking u_2 as an example, it is updated by the messages propagated from its neighbors (*i.e.*, u_1 and u_3).

we introduce the design of the spatial graph neural network for skeleton feature extraction.

As shown in Fig. 3, we select J key joints (*i.e.*, head, spine, left and right shoulders, left and right elbows, left and right wrists, left and right hands), where $J = 10$. We encode all the joints with 3D coordinates into a frame-level skeletal representation using a graph neural network. Different from concatenating all the 3D coordinates as a vector [18], [19] or tackling the spatial distribution as an image (*e.g.*, extracting 'visual' features from skeletal distribution images by CNNs [20]), we learn the skeletal relation using the graph convolutional network (GCN) [55]. We take the selected joints as nodes and build an incomplete undirected graph according to body connectivity. Specifically, if the input joint data of the n-th frame is denoted as $u_n \in \mathbb{R}^{J\times3}$, the adjacency matrix of the joints $\widehat{\mathbf{A}} \in \mathbb{R}^{J\times J}$ is elaborated in Fig. 3. $\widehat{\mathbf{A}}$ describes body connectivity, and its matrix sparsity reduces the computational cost in message passing. The update of node u_i with neighbors $\{u_j\}$ is conducted by a GCN operation as follows:

$$\begin{cases} \lambda(u_i) : u_i \to \{u_j | u_i u_j \in \widehat{\mathbf{A}}\}; \\ u_i' = \sum\limits_{u_j \in \lambda(u_i)} \frac{1}{\|\lambda(u_i)\|} \sigma(W_{j\to i} \times u_j), \end{cases} \tag{1}$$

where $\lambda(u_i)$ represents the neighbor set of u_i, the term $\|\lambda(u_i)\|$ denotes the number of neighbors, and W is a to-be-learned parameter. The proposed Skeleton-GCN is conducted by two-layer graph convolution (Eq. 1) and a fully connected (FC) layer, which is formulated as follows:

$$\begin{aligned} &\mathcal{V}^s = Skeleton-GCN(\{u_n\}|_{n=1}^N, \widehat{\mathbf{A}}) \Leftrightarrow \\ &\begin{cases} u_n' = ReLU(GCN(\widehat{\mathbf{A}} u_n W^1)); \\ u_n'' = ReLU(GCN(\widehat{\mathbf{A}} u_n' W^2)); \\ \mathcal{V}^s = \{v_n^s\}_{n=1}^N = FC(\{u_n''\}) \in \mathbb{R}^{N\times d}, \end{cases} \end{aligned} \tag{2}$$

where $W_1 \in \mathbb{R}^{3\times12}$ and $W_2 \in \mathbb{R}^{12\times48}$ are two learnable parameters, and $d = 512$.

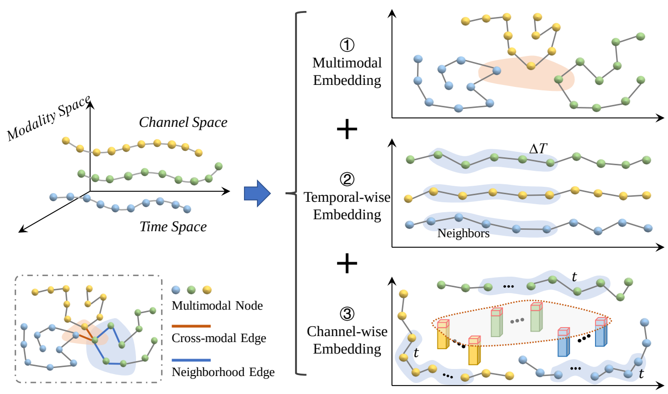


Fig. 4. The core idea of the GEU. t denotes the number of consecutive frames in the sampled clip, and ΔT represents the window size for constructing neighborhood edges (here, $\Delta T = 5$).

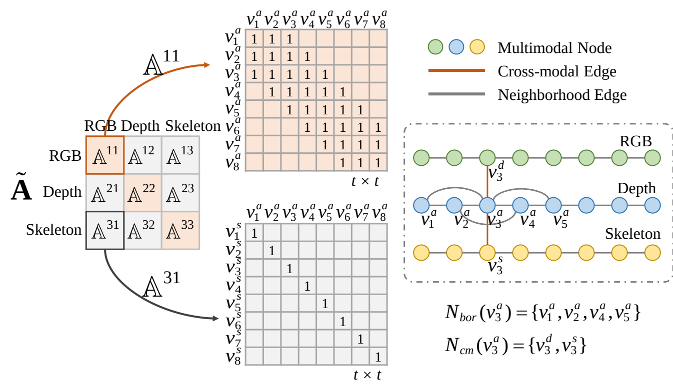


Fig. 5. Intra- and inter- modality correlation in the GEU module.

B. Multimodal Sequential Embedding

To date, we have extracted independent multimodal feature sequences, *i.e.*, $\mathcal{V}^a$, $\mathcal{V}^d$, and $\mathcal{V}^s$. To obtain a better representation of the video, we explore the implicit spatiotemporal cues and the relationship among multi-modalities. To this end, we propose the MSeqGraph model to explore the spatiotemporal cues and modal correlation of multimodal sequential features in a graph stack architecture. We first elaborate on the graph embedding unit (GEU) in MSeqGraph and then introduce the hierarchical GEU stack for SLT.

1) Graph Embedding Unit (GEU): As shown in Fig. 2, the GEU module consists of parallel CNN embedding and multimodal graph embedding. Based on multimodal feature sequences $\mathcal{V}^a$, $\mathcal{V}^d$, and $\mathcal{V}^s$, we attempt to learn short-term temporal relations among several adjacent frames. We sample t continuous frames from m feature sequences, where each modality feature is fixed to d-dim in Sec. III-A. Thus, we obtain a clip-level feature map $\mathcal{F} \in \mathbb{R}^{m\times t\times d}$. In our work, here are $m = 3$, $t = 8$, and $d = 512$. Then, a parallel CNN operation (PCN) is designed to model temporal-wise correlation (PCN_T) and channel-wise learning (PCN_C) of the feature map $\mathcal{F} \in \mathbb{R}^{m\times t\times d}$ as follows:

$$\mathcal{H} = PCN(\mathcal{F}) \in \mathbb{R}^{m\times t\times d} \Leftrightarrow \\ \begin{cases} \mathcal{F}_{tem} = PCN_T : ReLU(BN(Conv3D(\mathcal{F})))|_{kernel=(1,3,1)}; \\ \mathcal{F}_{cha} = PCN_C : ReLU(BN(Conv3D(\mathcal{F})))|_{kernel=(1,1,3)}; \\ \mathcal{H} = [\mathcal{F}_{tm} \oplus \mathcal{F}_{cha}], \end{cases} \tag{3}$$

where BN denotes the *BatchNorm* operation and $\oplus$ is the element-wise addition. Each vector in feature map $\mathcal{F} \in \mathbb{R}^{m\times t\times d}$ is transformed into a new vector in $\mathcal{H} \in \mathbb{R}^{m\times t\times d}$.

Next, we explore the cross-modal correlation. As shown in Fig. 5, we construct a multimodal graph G, which contains $m \times t$ nodes. Each node is the feature vector in $\mathcal{H} \in \mathbb{R}^{m\times t\times d}$. Observing Fig. 5, in our work, the intra-modality correlation is performed with a window $\Delta T = 5$, *i.e.*, the neighboring edges; the inter-modality correlation is conducted at the same time step, *i.e.*, the cross-modal edges. Thus, we set the adjacency matrix $\tilde{\mathbf{A}} \in \mathbb{R}^{(m\cdot t)\times(t\cdot m)}$ as follows:

$$\widetilde{\mathbf{A}} = \begin{bmatrix} \mathbb{A}^{11} & \mathbb{A}^{12} & \mathbb{A}^{13} \\ \mathbb{A}^{21} & \mathbb{A}^{22} & \mathbb{A}^{23} \\ \mathbb{A}^{31} & \mathbb{A}^{32} & \mathbb{A}^{33} \end{bmatrix} \in \mathbb{R}^{(m\cdot t)\times(t\cdot m)}. \tag{4}$$

$\widetilde{\mathbf{A}}$ contains two types of relations: the intra-modality correlation $\mathbb{A}^{ii}$ corresponding to neighboring edges with window ΔT in each modality itself, and the inter-modality correlation $\mathbb{A}^{ij}$ with $m = 3$ modalities (corresponding to cross-modal edges at the same time). Both $\mathbb{A}^{ii}$ and $\mathbb{A}^{ij}$ belong to diagonal matrices. $\mathbb{A}^{ij}$ is an identity matrix, *i.e.*, $\mathbb{A}^{ij} = \mathbf{I}_{t\times t} \in \mathbb{R}^{t\times t}$. $\mathbb{A}^{ii} \in \mathbb{R}^{t\times t}$ is a diagonal matrix with ΔT diagonals that is formulated in Eq. 5. For example, while $t = 8$ and $\Delta T = 5$, $\mathbb{A}^{11}$ is given in Fig. 5.

$$\mathbb{A}^{ii} = \begin{bmatrix} a_{1,1} & \cdots & a_{1,\lceil \frac{\Delta T}{2} \rceil} & 0 & 0 & 0 \\ a_{2,1} & a_{2,2} & \cdots & a_{2,\lceil \frac{\Delta T}{2} \rceil+1} & 0 & 0 \\ 0 & 0 & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & a_{t,t-\lfloor \frac{\Delta T}{2} \rfloor} & \cdots & a_{t,t} \end{bmatrix}_{t\times t} \tag{5}$$

After modeling $\widetilde{\mathbf{A}}$, we adopt the GCN operation to update intra-modal and inter-modal relations among nodes. Based on the input feature map $\mathcal{F} \in \mathbb{R}^{m\times t\times d}$, the graph embedding process is formulated to learn the node representation as follows:

$$\mathcal{H}' = GEU(\mathcal{F}, \widetilde{\mathbf{A}}) \in \mathbb{R}^{m\times t\times \frac{d}{2}} \Leftrightarrow \\ \begin{cases} \mathcal{H} = PCN(\mathcal{F}); \\ \mathcal{H}' = ReLU(GCN(\widetilde{\mathbf{A}}\mathcal{H}W_g)), \end{cases} \tag{6}$$

where $W_g \in \mathbb{R}^{d\times \frac{d}{2}}$ is a to-be-learned parameter. Thus, $\mathcal{F} \in \mathbb{R}^{m\times t\times d}$ is transformed into $\mathcal{H}' \in \mathbb{R}^{m\times t\times \frac{d}{2}}$, strengthening the modal complementarity and temporal correlation.

2) A Hierarchical GEU Stacker: Motivated by the fact that a deeper neural network improves model performance by inhibiting the network degradation [6], we design a hierarchical GEU stacker. The stacking details are depicted in Figs. 2 and 8. We set the three-layer GEU stack. In addition, there are two stacking modes in each GEU module, as shown in Fig. 6; we select the second mode, *i.e.*, embedding the skip connections operation into the GEU stacker. We also discuss

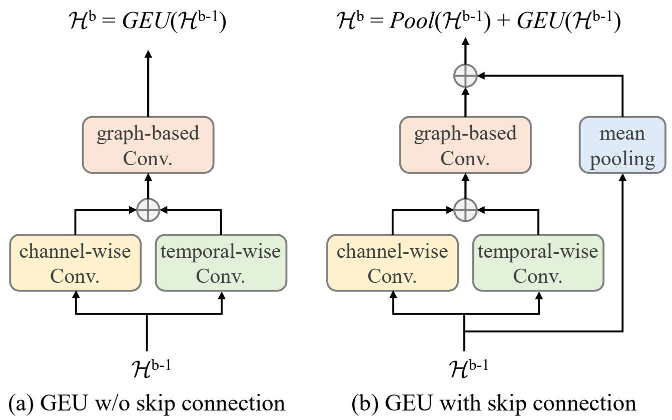


Fig. 6. Illustration of the GEU with a skip connection.

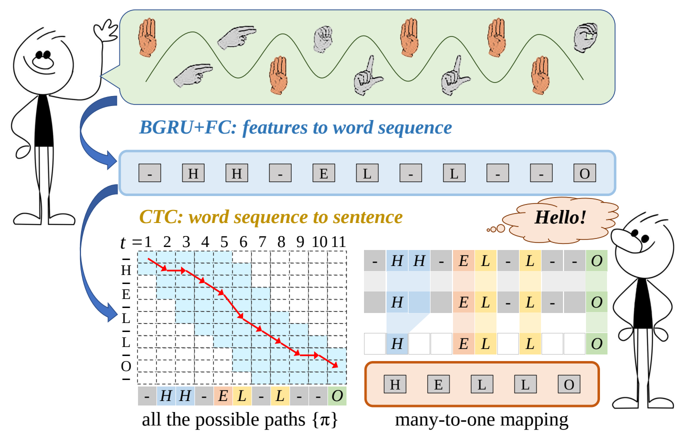


Fig. 7. Illustration of connectionist temporal decoding. In the training process, the CTC optimizer calculates all the possible paths $\{\boldsymbol{\pi}\}$. During the testing process, we pick up the path with the maximum probability score, *e.g.*, the red path in this figure, and apply many-to-one mapping decoding to statically generate a result. Note that to simplify this process, we show character-level decoding as our example; in reality, however, the decoding is conducted at the word level. In other words, the letters 'H,' 'E,' *etc.*, are changed to different words in our work.

the performances of these two modes in Sec. IV-B. Here, we rewrite the complete GEU stacker calculation in the proposed MSeqGraph as follows:

$$g = HGEU(\mathcal{F}, \widetilde{\mathbf{A}}) \Leftrightarrow \begin{cases} \mathcal{H}^b = \begin{cases} PCN(\mathcal{F}), & b = 0; \\ Pool(\mathcal{H}^{b-1}) + GEU(\mathcal{H}^{b-1}), & 1 < b \le B; \end{cases} \\ g = FC(\mathcal{H}^B) \in \mathbb{R}^{1\times \frac{d}{2^B}}, \end{cases} \tag{7}$$

where B is the height of the stacker.

To summarize, the hierarchical GEU stacker is designed to explore the short-term temporal cues in videos. For each video, we stack every 8 frames with 4-frame overlap to group K clips. Here, $K = \lfloor N/4 - 1 \rfloor$, where N is the frame number of a video. The feature map of each clip $\mathcal{F} \in \mathbb{R}^{m\times t\times d}$ is fed into the GEU stacker, and a fused feature vector $g \in \mathbb{R}^{1\times \frac{d}{2^B}}$ is output. Thus, we obtain a new embedding sequence of a video $\mathcal{G} = \{g_k\}_{k=1}^{K}$ through the GEU stacker.

C. Connectionist Temporal Decoding

In the decoding phase, the bidirectional GRU network (BGRU) and CTC model [56] are combined to jointly decode sentences. The BGRU-based CTC decoder used here includes a two-stage decoding and translation process. We use BGRU to realize sequential (temporal) learning, and CTC is adopted as the objective function to decode sentences. Specifically, we first explore longer-range temporal transitions across the entire video. The GEU stacker outputs are fed into the BGRU and the FC layer to map sequential features into a word vocabulary Voc.

$$\mathcal{P} = \{p_k\}_{k=1}^{K} = \varphi_{softmax}[FC(\{BGRU(g_k)\}_{k=1}^{K})] \tag{8}$$

where $\mathcal{P} = \{p_k\}_{k=1}^{K} \in \mathbb{R}^{K\times|Voc|}$ is a score matrix and $|Voc|$ is the size of Voc. We denote Voc as a set of all the words in the training set and add a blank word '_' to it.

The CTC optimizer applies a many-to-one mapping operation $\mathcal{B}$, as shown in Fig. 7, which merges the repetitions and deletes the blank words in path $\boldsymbol{\pi}$, *e.g.*, $\mathcal{B}(\boldsymbol{\pi})$ = $\mathcal{B}$ (_ *H H* _ *E L* _ *L* _ _ *O*) = (_ *H* _ *E L* _ *L* _ *O*)= $\{HELLO\}$. $\boldsymbol{\pi}$ is converted into a variable sentence $\mathcal{Y} = \{HELLO\}$. Therefore, actually, the probability of a labeling $\mathcal{Y} = (y_1, y_2, ..., y_L)$ containing L words is the probability sum of all the possible $\{\boldsymbol{\pi}\}$ with K probabilities p_k as follows:

$$\Pr(\mathcal{Y}|p_k) = \sum_{\boldsymbol{\pi}_k \in \mathcal{B}^{-1}(\mathcal{Y})} \Pr(\boldsymbol{\pi}_k|p_k) \tag{9}$$

where $\mathcal{B}^{-1}(\mathcal{Y}) = \{\boldsymbol{\pi}|\mathcal{B}(\boldsymbol{\pi}) = \mathcal{Y}\}$ involves all the possible paths $\{\boldsymbol{\pi}\}$. The probability of a path $\boldsymbol{\pi}$ is defined as follows.

$$\Pr(\boldsymbol{\pi}|p_k) = \prod_{k=1}^{K} \Pr(\pi_k|p_k), \forall \pi_{k,j} \in Voc' \tag{10}$$

where π_k is the k^{th} element of $\boldsymbol{\pi}$.

CTC optimization is regarded as maximizing the probability of all alignments; thus, the loss function is formulated as follows:

$$\mathcal{L} = \sum_{\pi\in\mathcal{B}^{-1}(\mathcal{W})} -log\mathcal{P}^{\pi} = -\sum_{\pi\in\mathcal{B}^{-1}(\mathcal{W})} \sum_{k=1}^{K} p_k^{\pi}. \tag{11}$$

In the test decoding, we obtain the probability score $\mathcal{P} = \{p_k\}_{k=1}^{K}$. Next, we use the *argmax* function on p_k and output the i^{th} word classification label with the maximum value. Finally, we have to merge the reduplicate words and delete the blank '_' by the above-mentioned many-to-one mapping $\mathcal{B}$, and output the final generated sentence.

IV. Experiments

A. Experimental Setup

1) Dataset: We evaluated the proposed MSeqGraph model on two benchmarks: USTC-CSL [12] and BOSTON-104 [57]. As shown in detail in Table I and Fig. 9, USTC-CSL is a Chinese sign language dataset that covers 100 daily sentences played by 50 signers. Referring to [10], we adopt two strategies, *Split I* and *Split II*, to split the dataset into training and testing sets. *Split I* is designed for the signer independent test, in which the sentences of training and testing sets are the

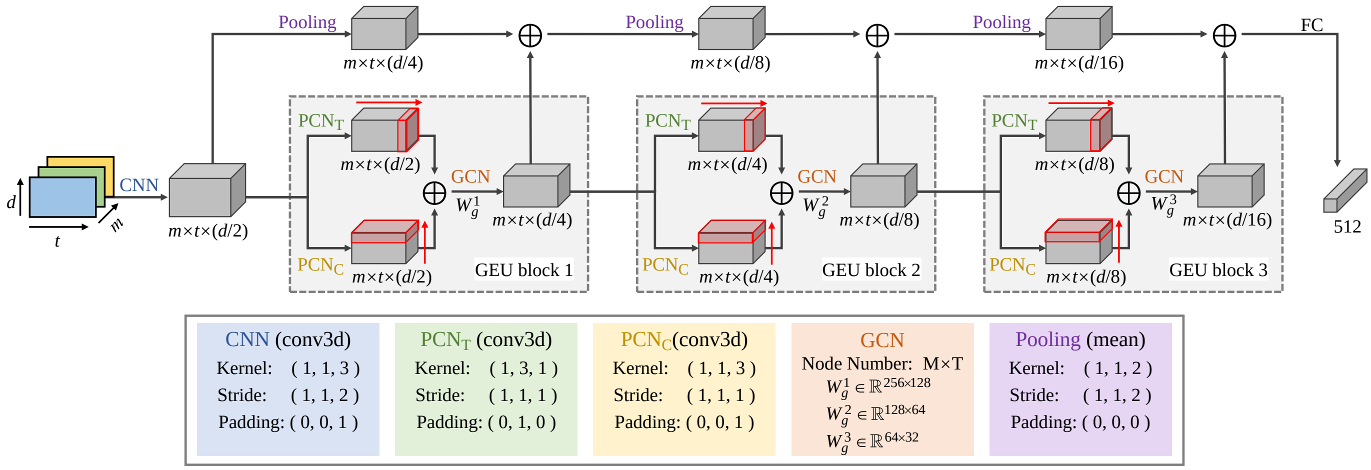


Fig. 8. Implementation details of the proposed MSeqGraph for multimodal sequential embedding. Given an input (clip-level feature map) $\mathcal{F} \in \mathbb{R}^{m \cdot t \cdot d}$, in each graph block (Graph Embedding Uint, GEU), PCN is designed for temporal-wise and channel-wise learning in parallel and GCN blends multimodal complementarity. Pooling-based skip connections linearly stack nonlinear GEUs. Different from the state-of-art models [15], [35] compacting temporal cues (transforming several frames into a clip, *i.e.*, $t \rightarrow \frac{t}{16}$), in this paper, we densely compact the channels to obtain $F' \in \mathbb{R}^{m \cdot t \cdot \frac{d}{16}}$, where the dimension reduction of feature maps is the same.

TABLE I
DETAILS OF BENCHMARK DATASETS

Split Strategies		Signers	Sentences	Videos	Vocabulary
USTC-CSL					
Split I	Train	40	100	4000	178
	Test	10	100	1000	178
Split II	Train	50	94	4700	178
	Test	50	6	300	20
BOSTON-104					
Train		3	122	161	103
Test		3	35	40	65

same but played by different signers; *Split II* evaluates the unseen sentence translation test, in which each word in the testing set exists in the training set, but the order of occurrence and the usage are completely different. BOSTON-104 contains 201 sentences of American sign language, referring to a vocabulary of 104 words. In BOSTON-104, 26% of the vocabulary words occur only once in the training corpus. It is noteworthy that our method focuses on solving the multimodal sequential embedding in SLT and translates sentences. Thus, the famous RGB-based single-modal datasets PHOENIX14 [7] and PHOENIX14T [42] are not considered.

2) Evaluation Metrics: *WER* (word error rate) [7] is used to measure the similarity of two sentences, which is calculated as $WER = \frac{DEL+INS+SUB}{num\_words}$, where num_words stands for the number of words in the ground-truth, and *DEL*, *INS*, *SUB* denote the numbers of deletions, insertions and replacements with the minimum total operations during the transformation of the generated sentence into the ground-truth. *Precision* is the ratio of correct sentences to all the sentences. *Acc-w* is the average ratio of correct words in each generated sentence to the corresponding ground-truth. In addition, we adopt the semantic metrics used in the fields of NLP [58], NMT [59] and image captioning [60], such as *CIDEr*, *BLEU*, *ROUGE-L*

(a) An video example of USTC-CSL with RGB images, depth frames and skeletal data

(b) An video example of BOSTON-104 with black-and-white images

Fig. 9. Video examples of (a) USTC-CSL and (b) BOSTON-104. The skeleton data of USTC-CSL is captured by Kinect V2.0. In BOSTON-104, black-and-white images are annotated with positions of hands and face.

and *METEOR*.

3) Implementation Details: For feature extraction of RGB and depth images, we adopt a pretrained ResNet-18 on ImageNet [61], where the images are cropped with a size of 224×224, and output through the average pooling layer after $conv5\_x$ of ResNet-18, where the feature dimension is set to 512. For skeleton data in the USTC-CSL dataset, we select 10 key joints (*i.e.* head, spine, left and right shoulders, left and right elbows, left and right wrists, left and right hands) with three-dimensional coordinates collected by Kinect V2.0. For dataset BOSTON-104, as shown in Fig. 9 (b), 2D-dim positions (x,y) of hands and face are leveraged. Using the proposed Skeleton-GCN model, we obtain the respective 512-dim skeleton feature sequence for the USTC-CSL and BOSTON-104 datasets. Note that for the data stream of

TABLE II
EVALUATION OF DIFFERENT MODALITY SETTINGS

Features	WER(%)↓	CIDEr↑	BLEU-1↑	ROUGE-L↑	METEOR↑
Experimental Results on *Split I*					
RGB	17.9	7.364	0.848	0.853	0.537
Depth	14.8	6.788	0.852	0.879	0.512
Skeleton-concat	12.7	7.413	0.907	0.907	0.559
Skeleton-MLP	11.0	7.650	0.897	0.908	0.566
Skeleton-GCN	8.5	8.520	0.938	0.938	0.630
RGB+Depth+Skeleton-GCN	**6.3**	**9.020**	**0.942**	**0.958**	**0.653**
Experimental Results on *Split II*					
RGB	63.3	0.503	0.472	0.479	0.182
Depth	61.5	0.571	0.411	0.444	0.158
Skeleton–concat	62.5	0.604	0.466	0.469	0.196
Skeleton-MLP	61.7	0.504	0.474	0.468	0.183
Skeleton-GCN	59.5	0.627	**0.493**	0.485	**0.201**
RGB+Depth+Skeleton-GCN	**59.1**	**0.705**	0.467	**0.498**	**0.201**

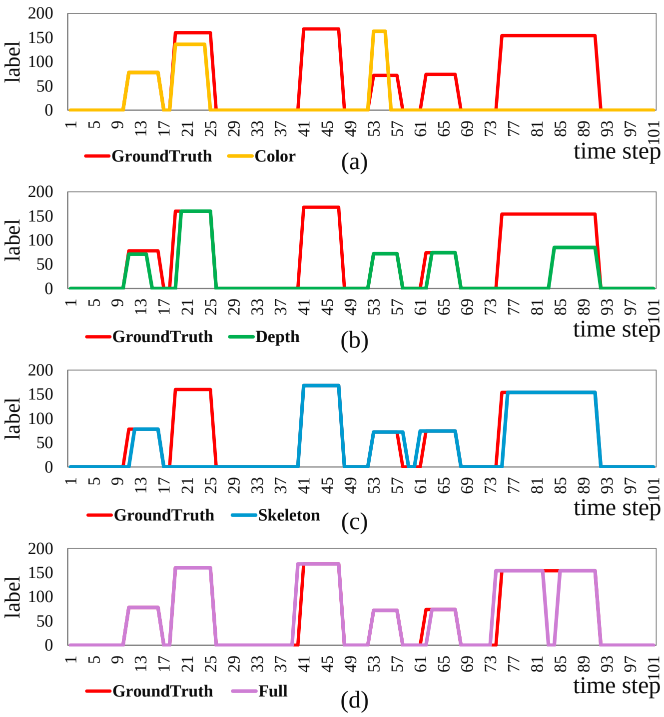


Fig. 10. Visualization of word-level classification accuracy of a USTC-CSL video example. The X-axis represents the time step, and the Y-axis stands for word labels, where label #0 denotes the 'blank' action.

videos, we sequentially sample every eight features with four-frame overlap as clip units and feed them into the proposed MSeqGraph. The detailed modules of the proposed MSeqGraph are shown in Fig. 8. In addition, we apply batch normalization [62] after each convolutional layer, and BGRU with $2\times1024-dim$ hidden states for CTC decoding. We adopt the ADAM [63] optimizer and set the batch size to 20. The learning rate is initially set to 1×10^{-4} and then set to 1×10^{-5} after 20 epochs. The model finally achieves the convergence after approximately 60 epochs of training. Experiments are performed with PyTorch on NVIDIA GeForce GTX 1080 Ti GPU.

B. Ablation Studies

1) Experiments with Multimodal Cues: As shown in Table II, skeleton features perform more robust *WER* than

TABLE III
EVALUATION OF THE GRAPH EMBEDDING UNIT

Structures	WER(%)↓	CIDEr↑	BLEU-1↑	ROUGE-L↑	METEOR↑
Experimental Results on *Split I*					
GEU w/o C	4.6	9.048	0.963	0.966	0.678
GEU w/o T	4.5	9.096	0.960	0.966	0.678
GEU w/o G	5.3	8.890	0.956	0.959	0.672
GEU w/o Skip	4.4	8.832	0.959	0.966	0.662
w/o GEU	6.3	9.020	0.942	0.958	0.653
Intact *GEU*	**0.6**	**9.666**	**0.995**	**0.995**	**0.810**
Experimental Results on *Split II*					
GEU w/o C	60.3	0.479	0.451	0.430	0.172
GEU w/o T	56.8	0.628	0.480	0.490	0.200
GEU w/o G	60.9	0.584	0.468	0.480	0.176
GEU w/o Skip	57.5	0.672	0.436	0.488	0.174
w/o GEU	59.1	0.705	0.467	0.498	0.201
Intact *GEU*	**49.9**	**1.061**	**0.531**	**0.566**	**0.234**

RGB and depth features. The experimental results for the *CIDEr*, *BLEU-1*, *ROUGE-L* and *METEOR* metrics also verify this conclusion. This indicates that learning visual cues is more difficult than learning skeletal data for sign language recognition. In addition, there is an interesting phenomenon in which visual features of depth images perform better on *WER* but worse on the other semantic metrics than RGB features. For example, the *CIDEr* of depth features on *Split II* increases more than 10% compared with RGB features. Note that *WER* indicates the incorrectly identified words, whereas *CIDEr*, *BLEU-1*, *ROUGE-L* and *METEOR* demonstrate the semantic measurement. This reflects that RGB features can identify synonyms but have difficulty to solve the only correct one; skeleton and depth data seem to be more powerful at distinguishing the correct words. For skeletal data, to verify the proposed Skeleton-GCN module, we compare with the method concatenating 3D coordinates as a vector, denoted as **Skeleton-concat**. The *WERs* of **Skeleton-GCN** are 4.2 / 3.0 better than **Skeleton-concat** on *Split I* and *Split II*. We further set a variant of **Skeleton-GCN** - **Skeleton-MLP**, which implements MLP on the body joints' coordinate data rather than graph modeling. Compared with other single-modal methods, **Skeleton-GCN** achieves the best performance, and its *BLEU-1* and *METEOR* on *Split II* have even caught up or surpassed the respective values for the multimodal method. These results indicate that GCN further strengthens the performance advantages of skeleton features, especially in semantic similarity evaluation.

In addition, we visualize the alignment of the respective feature sequence and words of a video sample in Fig. 10, where the red line records the ground-truth. It is observed that both RGB and depth features obviously miss a matching at time steps 41∼49. RGB performs the worst at time steps 75∼93. Skeleton data perform well most of the time except for an obvious missing at steps 17∼27. Regardless, the combination of all the modality data performs the best. In Fig. 10(d), although there are several skipping frames, they do not affect the generated words through the greedy merging statically in the CTC decoding path. More qualitative results are given in Fig. 11. The combination of all the modality data improves the sentence prediction.

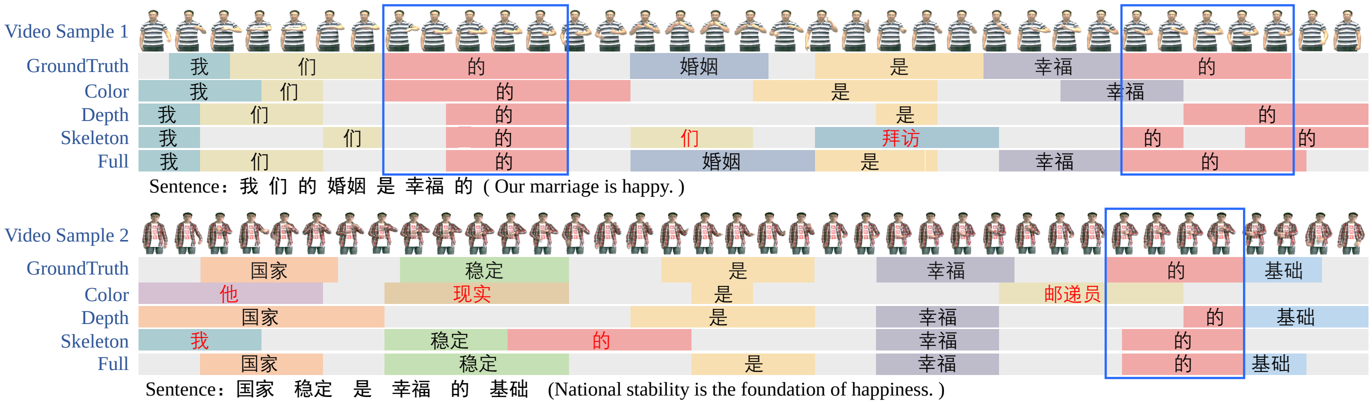
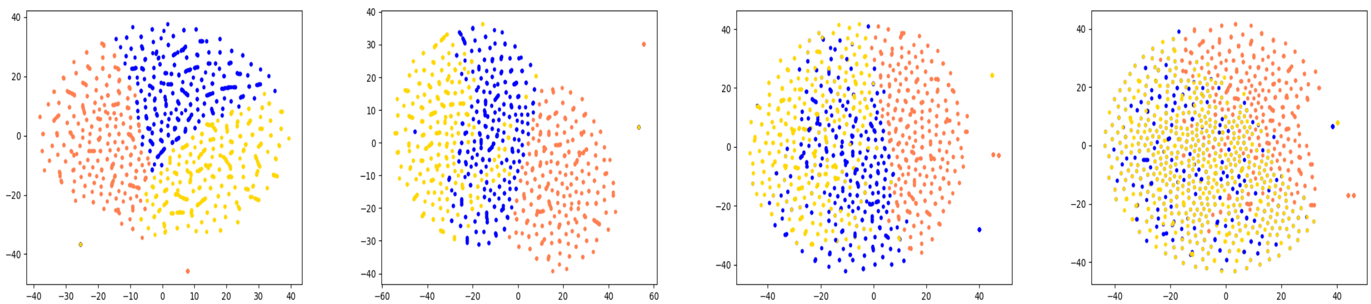


Fig. 11. Visualization translation examples. Each colored block marks the respective generated word. The gray block is in accord with the 'blank' label, and the red word denotes an incorrectly predicted word. The blue border marks the temporal boundary of the form word.



(a) Original features (b) GEU block 1 (c) GEU block 2 (d) GEU block 3

Fig. 12. Embedding visualization of the hierarchical GEU stacker of two USTC-CSL video examples displayed by t-SNE. Red, yellow, and blue points represent skeleton, depth, and RGB features, respectively. In each feature space, skeleton features derived from coordinates (*x, y, z*) perform differently from the other two; RGB features and depth features gradually draw closer together as they are extracted as two types of visual features.

2) Evaluation of the Graph Embedding Unit (GEU): We define several variants of GEU and test them to verify the effectiveness of GEU: ***GEU w/o C*** (removing $\mathcal{F}_{cha}$ in Eq. 3), ***GEU w/o T*** (removing $\mathcal{F}_{tm}$ in Eq. 3), ***GEU w/o G*** (removing $GCN(\cdot)$ in Eq. 6) and ***GEU w/o Skips*** (removing $Pool(\cdot)$ in Eq. 7). Among all the variants of GEU, the worst performance occurs on ***GEU w/o G***. Compared with **Intact *GEU***, the *WER* of ***GEU w/o G*** increases 4.7 and 11.0 with *Split I* and *Split II*. This indicates that without relational learning by GCN, the capability of the model to capture multimodal complementarity weakens rapidly. The second worst performance occurs on ***GEU w/o C***, especially on *Split II* (*i.e.*, *WER* +10.4, *CIDEr* -0.582, *ROUGE-L* -0.136 compared with **Intact *GEU***), which indicates that channel-wise learning is crucial to capture the sign semantics. Compared with ***GEU w/o Skips***, the *WER* of **Intact *GEU*** drops 4.4 / 57.5 to 0.6 / 49.9 with *Split I* and *Split II*, which verifies the positive impact of skip connections to inhibit the network degradation as described in [6] (as shown the discussion in Sec. III-B).

In addition, ***w/o GEU*** denotes using an eight-layer multi-layer perceptron (MLP) to replace all the ***GEU*** modules in the proposed MSeqGraph, which shows the worst performance in Table III, *e.g.*, *WER* and *METEOR* of ***w/o GEU*** on *Split I* are worse (+5.7 and -0.157) than **Intact *GEU***. It demonstrates the merit of the ***GEU***.

TABLE IV
EVALUATION OF GEU BLOCKS IN THE GEU STACKER

Blocks	WER(%)↓	CIDEr↑	BLEU-1↑	ROUGE-L↑	METEOR↑
Experimental Results on *Split I*					
B = 0	6.3	9.020	0.942	0.958	0.653
B = 1	4.1	8.997	0.966	0.965	0.690
B = 2	1.7	9.493	0.986	0.989	0.755
B = 3	**0.6**	**9.666**	**0.995**	**0.995**	**0.810**
B = 4	5.0	8.497	0.950	0.962	0.633
Experimental Results on *Split II*					
B = 0	59.1	0.705	0.467	0.498	0.201
B = 1	57.1	0.588	0.458	0.496	0.181
B = 2	52.4	0.885	0.501	0.552	0.211
B = 3	**49.9**	**1.061**	**0.531**	**0.566**	**0.234**
B = 4	62.5	0.458	0.436	0.433	0.171

3) Evaluation of GEU Blocks: We test the effect of the GEU on feature embedding. As shown in Table IV, *WER* achieves the best when three GEU blocks are stacked. We set the empirical parameter of $B = 3$. The t-SNE visualization in Fig. 12 shows the feature distribution of each GEU block. We randomly select a batch of samples of the testing set on USTC-CSL *Split I*. As shown in Fig. 12(a), the distributions of the original modalities are separated from each other in the feature space; in Fig. 12(b), (c) and (d), RGB and depth features are aggregated into a close spatial distribution, where

TABLE V
EVALUATION OF THE GEU AND ALTERNATIVES ON USTC-CSL

Modules	WER(%)↓	CIDEr↑	BLEU-1↑	ROUGE-L↑	METEOR↑
Experimental Results on *Split I*					
MLP	6.3	9.020	0.942	0.958	0.653
MFB	5.9	8.778	0.947	0.953	0.658
LMF	5.0	8.823	0.962	0.961	0.661
GEU	**0.6**	**9.666**	**0.995**	**0.995**	**0.810**
Experimental Results on *Split II*					
MLP	59.1	0.705	0.467	0.498	0.201
MFB	58.8	0.495	0.429	0.468	0.147
LMF	62.6	0.495	0.474	0.473	0.173
GEU	**49.9**	**1.061**	**0.531**	**0.566**	**0.234**

skeleton features perform very differently. This is attributable to the characteristics of multimodal data in which the RGB and depth features belong to visual cues and are even extracted by the same ResNet-18 model [6]. In contrast, skeleton features are extracted from coordinate data by the proposed Skeleton-GCN model.

4) Evaluation of the GEU and Alternatives: We compare the **GEU** with some existing alternatives, such as **MLP**, **MFB** [46] and **LMF** [48]. Multi-layer perceptron (**MLP**) is a simple but effective deep model, whose performance can be regarded as a baseline for reference. Multimodal factorized bilinear (**MFB**) [46] factorizes the projection matrix into two low-rank matrices and designs the bilinear pooling with a co-attention mechanism to aggregate multimodal features. Low-rank multimodal fusion (**LMF**) [48] decomposes the multi-modal mapping weight into a set of modality-specific low-rank factors, so that the fusion output can be directly computed without explicitly tensorizing the unimodal representations.

We replace the proposed **GEU** module with **MLP**, **MFB** and **LMF** respectively, and the results are shown in Table V. Since **MFB** and **LMF** were originally proposed to tackle dual-stream fusion, they performed well at the correlation calculation of cross-modal vector pairs. However, compared with **MLP**, the performances of **MFB** and **LMF** are not significantly improved, and even **LMF** (*WER* 62.6) performs far worse than others on *Split II*. Similarly, although **MFB** achieves *WER* to 58.8 on *Split II*, it lacks advantages on *Split I*. In contrast, the **GEU** fuses feature streams by graph-based embedding, which enables complementary cues to fully interact among multiple modalities. Thus, our method has significant advantages in all metrics and maintains robustness over both split tasks.

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

We compare the proposed model MSeqGraph with the existing approaches: LSTM&CTC [64], ELM [69], S2VT [65], HRNN [68], HRNE [66], MFB [46], CTF [11], HLSTM [10], WIC-NGC [67], CTM [13], HRF [14], PTE [70], LMF [48] and KA-JointCTC [15] on the USTC-CSL dataset; MLP, MFB [46], LMF [48], SRT [71], EA [72], VMFA [73], SDM [74], PTE [70] and CTM [13] on the BOSTON-104 dataset.

1) Experiments on USTC-CSL: The experimental results are listed in Table VI. The classic encoder-decoder framework is widely used in sign language translation, such as **S2VT** [65], which is a classic encoder-decoder model based on two-layer LSTMs and the expanded version **S2VT-3L** [65], hierarchical RNN - **HRNE** [66], **HRNN** [68] and hierarchical LSTM - **HLSTM** [10]. **LSTM&CTC** [64] is another classic framework with LSTM encoding and CTC decoding. **KA-JointCTC** [15] proposes a pyramid BiLSTM to encode key actions, and aggregates both CTC decoding and LSTM decoding to generate the sentence. **WIC-NGC** [67] designs multiple classifiers; each classifier outputs only one word or n-gram phase and all the outputs combine a sentence. Among the above methods, **WIC-NGC** achieves the best performances of *WER* 50.9, *BLEU-1* 0.505, *GOUGE-L* 0.537 on *Split II*. Compared with the above mentioned works, we embed relational graph learning to optimize the feature embedding phase. The proposed **MSeqGraph** achieves the best performances, including dropping *WER* by 1.0 compared to **WIC-NGC** on *Split II*. The obvious performances are shown on *Split I*.

We also compare with some typical fusion methods, such as S2VT-Fusion [65], CTM [13], ELM [69], CTF [11] and HRF [14]. **CTM** [13] adopts element-wise summation as the fusion strategy. **ELM-Early** [69] directly concatenates features and **ELM-Late** [69] fuses probability scores from multiple ELM models. **CTF** [11] explores feature fusion and score fusion. In contrast, we aggregate multimodal cues by graph learning. We employ the graph-based stack (GEU-based stacker) to model multimodal correlation for feature embedding. **MSeqGraph** performs the best on USTC-CSL *Split I*, *e.g.*, *WER* 0.6, *CIDEr* 9.666, *BLEU-1* 0.995, *GOUGE-L* 0.995. This indicates that learning cross-modal complementarity in a gradual aggregation manner takes effect. In addition, USTC-CSL *Split II* offers a challenging task by which to evaluate the unseen sentence translation. Even though, as shown in Table VI, our work performs better than the others, especially within *CIDEr* 1.061 and *ROUGE-L* 0.566 on *Split II*.

We further compare with some fusion methods (**MFB** [46], **LMF** [48], **CTM-Fusion** [13], and **PTE** [70]) that use the same inputs as our approach. Here, **MFB** [46] is used to embed bilinear pooling into a co-attention mechanism for multimodal fusion. **LMF** [48] leverages low-rank weight tensors to make multimodal fusion efficient, which achieves the best *WER* 5.0 on *Split I* among the compared fusion methods. Our graph-based GEU module considers a wider temporal range in one calculation, which takes all the multimodal sequential frames in a clip as nodes and infers the correlation of all the nodes in the graph. As an extension of CTM [13], **CTM-Fusion** first independently learns short-term temporal cues in each modality, and then weights summed multimodal features for sentence translation, which improves *WER* from 61.9 to 52.7 on *Split II* compared to CTM. **PTE** [70] proposes parallel CNN and LSTM to encode and concatenate multimodal features. Compared with the simple fusion operation (*e.g.* concatenating or summation) in **PTE** and **CTM-Fusion**, our graph-based GEU module attempts to learn the advanced relation in the graph, leading to great improvement on *WER*, especially on *Split I*.

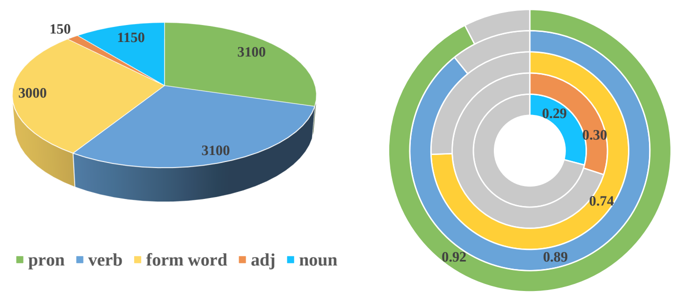
TABLE VI
PERFORMANCE COMPARISON ON THE USTC-CSL DATASET

Methods	Multimodal Inputs	*Split I* (Signer Independent Task)					*Split II* (Unseen Sentence Task)					
		WER(%)↓	CIDEr↑	BLEU-1↑	ROUGE-L↑	METEOR↑	WER(%)↓	Acc-w↑	CIDEr↑	BLEU-1↑	ROUGE-L↑	METEOR↑
LSTM&CTC [64]	RGB	11.9	8.632	0.936	0.940	0.646	75.7	0.332	0.241	0.343	0.362	0.111
S2VT [65]	RGB	25.5	8.512	0.902	0.904	0.642	67.0	0.457	0.479	0.466	0.461	0.189
S2VT-3L [65]	RGB	–	9.344	0.966	0.970	0.739	68.0	0.374	0.504	0.373	0.406	0.149
HRNE [66]	RGB	–	8.907	0.935	0.938	0.683	63.0	0.459	0.476	0.463	0.462	0.173
HLSTM [10]	RGB	10.7	9.019	0.942	0.944	0.699	66.2	0.482	0.561	0.485	0.481	0.193
KA-JointCTC [15]	RGB	9.1	–	–	–	–	59.4	–	–	–	–	–
WIC [67]	RGB	–	9.420	0.982	0.980	0.729	53.2	–	0.760	0.483	0.514	0.219
WIC-NGC [67]	RGB	–	9.416	0.979	0.979	0.725	50.9	–	0.641	0.505	0.537	0.223
CTF [11]	RGB	11.2	–	–	–	–	–	–	–	–	–	–
CTM [13]	RGB	–	–	–	–	–	61.9	–	–	–	–	–
HRNN [68]	skeleton	–	8.868	0.930	0.930	0.684	102.0	0.128	0.032	0.299	0.091	**0.279**
S2VT-Fusion [65]	RGB, skeleton	–	9.549	0.984	0.984	0.793	73.2	0.406	0.335	0.419	0.407	0.150
ELM-Early [69]	RGB, skeleton	–	8.101	0.874	0.874	0.559	96.8	0.367	0.240	0.348	0.352	0.116
ELM-Late [69]	RGB, skeleton	–	9.462	0.979	0.970	0.760	98.7	0.175	0.028	0.376	0.388	0.120
HRF [14]	RGB, skeleton	–	9.665	0.993	0.994	**0.817**	67.2	0.445	0.398	0.450	0.449	0.171
MFB [46]	RGB, depth, skeleton	5.9	8.778	0.947	0.953	0.658	58.8	0.410	0.495	0.429	0.468	0.147
CTM-Fusion [13]	RGB, depth, skeleton	8.1	8.316	0.928	0.936	0.615	52.7	0.477	0.869	0.486	0.513	0.223
LMF [48]	RGB, depth, skeleton	5.0	8.823	0.962	0.961	0.661	62.6	0.429	0.495	0.474	0.473	0.173
PTE [70]	RGB, depth, skeleton	14.6	6.343	0.837	0.893	0.496	58.9	0.314	1.053	0.388	0.500	0.168
Our Method	**RGB, depth, skeleton**	**0.6**	**9.666**	**0.995**	**0.995**	0.810	**49.9**	**0.485**	**1.061**	**0.531**	**0.566**	0.234

TABLE VII
PERFORMANCE COMPARISON ON THE BOSTON-104 DATASET

Methods	Input Data	DEL↓	INS↓	SUB↓	WER(%)↓
MLP	Frames+HP	23	14	22	34.95
MFB [46]	Frames+HP	20	8	18	27.37
LMF [48]	Frames+HP	15	13	15	26.85
SRT [71]	Frames+HP+HV+HT	–	–	–	17.90
EA [72]	Frames	40	9	18	30.34
VMFA [73]	PCA-Hand	–	–	–	28.65
SDM [74]	Frames+HT+PCA-Hand	**12**	8	15	19.66
PTE [70]	Frames+HP	35	6	9	28.47
CTM [13]	Frames+HP	32	17	11	36.74
Our Method	Frames+HP	18	**3**	**6**	**15.25**

'HP', 'HV' and 'HT' denote the features of hand-positions, hand-velocities and hand-trajectories respectively.



(a) Word frequency on *Split II-training set* (b) Word accuracy on *Split II-testing set*

Fig. 13. The distributions of word properties and recognition accuracies on the USTC-CSL dataset - *Split II*. Here, only the words appearing in the test set are counted.

Compared with the above methods, our method achieves better performance for the following reasons. First, existing methods (*e.g.*, S2VT [65], HRNE [66], HLSTM [10], WIC-NGC [67], CTF [11], CTM [13], and KA-JointCTC [15]) merely use visual data, while multi-source cues (*i.e.*, RGB/depth images and skeleton coordinates) are introduced into SLT in our method to eliminate the negative effect of data noise. Second, for multimodal methods (*e.g.*, S2VT-Fusion [65], ELM [69], and HRF [14]) tackling vision and skeleton data, the skeleton coordinates are concatenated into a vector. In contrast, we design a skeleton GCN to encode 3D coordinates, which fully explores the spatial relation among joints. Finally, existing fusion methods (*e.g.*, S2VT-Fusion [65], ELM [69], HRF [14], and CTM-Fusion [13]) usually regard multimodal fusion and sequence modeling as two independent processes, which ignores fine-grained cross-modal complementarity along the frame sequence. However, the proposed graph-based embedding simultaneously solves multimodal fusion and temporal learning problems, which provides more robust representations of sign videos.

2) Experiments on BOSTON-104: We compare our method with several typical multimodal embedding statics (*i.e.* **MLP**, **MFB** [46], **LMF** [48]); in this case, we remain the MSeqGraph framework unchanged, but modify the embedding or fusion modules. We also compare with the existing SLT works (**SRT** [71], **EA** [72], **VMFA** [73], **SDM** [74], **PTE** [70], and **CTM** [13]) to evaluate the sign language recognition performances on BOSTON-104. As shown in Table VII, our method achieves the best *WER*, which is 2.65 better than **SRT** [71] (the best result in the comparisons). In **EA** [72] and **VMFA** [73], the sign frames are cropped into a local area only covering the hands, whose performances are worse than most methods, especially on *DEL*. This observation indicates that in addition to the hand area, arm posture and facial expressions are also important for sign language recognition. **SRT** [71] and **SDM** [74] extract manual features from hand-position, hand-velocity and hand-trajectory data, which are weaker than the non-manual representations. **PTE** [70] and **CTM** [13] achieve better performances than other compared methods on the *SUB* metric, while their *DELs* are far worse than ours, which indicates that our method decodes fewer redundant words. In addition, fusion methods (*e.g.*, MLP, MFB [46], LMF [48], SRT [71], SDM [74], PTE [70], and CTM [13]) fuse multiple feature streams without temporal alignment. **MSeqGraph** extracts the multimodal features at the same level, and adopts graph-based embedding to learn fine-grained multimodal complementarity along the timeline, so as to obtain a more compact, complementary and informative sign representation.

D. Discussion of the Classification Accuracy of Part-of-Speech

To deeply investigate the model's word-level recognition accuracy, we conduct our test on the USTC-CSL dataset - *Split II*, which is more challenging with unseen sentences in real-world applications. As shown in Fig. 13, we display the distribution of word properties in the training subset and recognition accuracies of words in the testing subset. As shown in Fig. 13 (b), the recognized accuracy of pronouns, verbs, and form words is significantly higher than adjectives and nouns. This may be because pronouns, verbs, and form words frequently appear during training. However, nouns appear more frequently than adjectives, while their accuracy is lower. It seems that the complexity of nouns is more difficult to solve than adjectives. To summarize, while the distribution of words is unbalanced, this challenge needs to be explored.

V. CONCLUSION

In this paper, we propose a graph-based multimodal sequential embedding graph (MSeqGraph) network to solve sign language translation with multimodal cues. The proposed MSeqGraph model consists of channel-wise embedding, temporal-wise embedding, and multimodal relational embedding in a graph embedding unit (GEU). In the GEU, parallel channel-wise and temporal-wise convolutions are embedded into the GCN calculation. The GEU captures intra-modal and inter-modal modal complementarity by constructing temporal neighborhood edges and cross-modal edges. In addition, we exploit a hierarchical GEU stacker to further leverage dense multimodal cues. After that, we obtain a new integrated feature sequence along the temporal dimension from RGB, depth images, and skeletal data. We utilize the CTC optimizer to decode the sentence. Experiments on two benchmarks demonstrate the effectiveness of the proposed MSeqGraph and show that exploiting multimodal cues contributes to a better representation and improves performance.

REFERENCES

[1] K. L. Cheng, Z. Yang, Q. Chen, and Y.-W. Tai, "Fully convolutional networks for continuous sign language recognition," in *European Conf. Computer Vision (ECCV)*, 2020, pp. 697–714.

[2] N. C. Camgoz, O. Koller, S. Hadfield, and R. Bowden, "Sign language transformers: Joint end-to-end sign language recognition and translation," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 023–10 033.

[3] 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 *IEEE Int. Conf. Computer Vision (ICCV)*, 2015, pp. 2758–2766.

[4] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *AAAI Conf. Artificial Intelligence (AAAI)*, 2018, pp. 7444–7452.

[5] Z. Yang, Y. Xu, H. Wang, B. Wang, and Y. Han, "Multirate multimodal video captioning," in *ACM Int. Conf. Multimedia (ACM MM)*, 2017, pp. 1877–1882.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[7] O. Koller, S. Zargaran, and H. Ney, "Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4297–4305.

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

[9] J. Pu, W. Zhou, and H. Li, "Dilated convolutional network with iterative optimization for continuous sign language recognition," in *Int. Joint Conf. Artificial Intelligence (IJCAI)*, 2018, pp. 885–891.

[10] D. Guo, W. Zhou, H. Li, and M. Wang, "Hierarchical lstm for sign language translation," in *AAAI Conf. Artificial Intelligence (AAAI)*, 2018, pp. 6845–6852.

[11] S. Wang, D. Guo, W. Zhou, Z. Zha, and M. Wang, "Connectionist temporal fusion for sign language translation," in *ACM Int. Conf. Multimedia (ACM MM)*, 2018, pp. 1483–1491.

[12] J. Huang, W. Zhou, Q. Zhang, H. Li, and W. Li, "Video-based sign language recognition without temporal segmentation," in *AAAI Conf. Artificial Intelligence (AAAI)*, 2018, pp. 2257–2264.

[13] D. Guo, S. Tang, and M. Wang, "Connectionist temporal modeling of video and language: a joint model for translation and sign labeling," in *Int. Joint Conf. Artificial Intelligence (IJCAI)*, 2019, pp. 751–757.

[14] D. Guo, W. Zhou, A. Li, H. Li, and M. Wang, "Hierarchical recurrent deep fusion using adaptive clip summarization for sign language translation," *IEEE Trans. Image Process.*, vol. 29, pp. 1575–1590, 2020.

[15] H. Li, L. Gao, R. Han, L. Wan, and W. Feng, "Key action and joint ctc-attention based sign language recognition," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 2348–2352.

[16] R. Cui, H. Liu, and C. Zhang, "Recurrent convolutional neural networks for continuous sign language recognition by staged optimization," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1610–1618.

[17] O. Koller, S. Zargaran, H. Ney, and R. Bowden, "Deep sign: Enabling robust statistical continuous sign language recognition via hybrid cnn-hmms," *Int. J. Comput. Vis.*, vol. 126, no. 12, pp. 1311–1325, 2018.

[18] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1110–1118.

[19] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks," in *AAAI Conf. Artificial Intelligence (AAAI)*, 2016, pp. 3697–3703.

[20] Y. Du, Y. Fu, and L. Wang, "Skeleton based action recognition with convolutional neural network," in *IAPR Asian Conf. Pattern Recognition (ACPR)*, 2015, pp. 579–583.

[21] B. Wang, Y. Yang, X. Xu, A. Hanjalic, and H. T. Shen, "Adversarial cross-modal retrieval," in *ACM Int. Conf. Multimedia (ACM MM)*, 2017, pp. 154–162.

[22] D. C. Luvizon, D. Picard, and H. Tabia, "2d/3d pose estimation and action recognition using multitask deep learning," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 5137–5146.

[23] J.-C. Hou, S.-S. Wang, Y.-H. Lai, Y. Tsao, H.-W. Chang, and H.-M. Wang, "Audio-visual speech enhancement using multimodal deep convolutional neural networks," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 2, no. 2, pp. 117–128, 2018.

[24] T. Phan-Minh, E. C. Grigore, F. A. Boulton, O. Beijbom, and E. M. Wolff, "Covernet: Multimodal behavior prediction using trajectory sets," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 14 074–14 083.

[25] D. Guo, W. Zhou, H. Li, and M. Wang, "Online early-late fusion based on adaptive hmm for sign language recognition," *ACM Trans. Multimed. Comput. Commun. Appl.*, vol. 14, no. 1, pp. 1–18, 2017.

[26] H. Wang, A. Meghawat, L.-P. Morency, and E. P. Xing, "Select-additive learning: Improving generalization in multimodal sentiment analysis," in *IEEE Int. Conf. Multimedia and Expo (ICME)*, 2017, pp. 949–954.

[27] O. Kampman, E. J. Barezi, D. Bertero, and P. Fung, "Investigating audio, video, and text fusion methods for end-to-end automatic personality prediction," in *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018, pp. 606–611.

[28] J. Wan, G. Guo, and S. Z. Li, "Explore efficient local features from rgb-d data for one-shot learning gesture recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1626–1639, 2015.

[29] A.-A. Liu, N. Xu, W.-Z. Nie, Y.-T. Su, and Y.-D. Zhang, "Multi-domain and multi-task learning for human action recognition," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 853–867, 2018.

[30] R. Xie, C. Wang, and Y. Wang, "Metafuse: A pre-trained fusion model for human pose estimation," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 13 686–13 695.

[31] L. Sun, B. Li, C. Yuan, Z. Zha, and W. Hu, "Multimodal semantic attention network for video captioning," in *IEEE Int. Conf. Multimedia and Expo (ICME)*, 2019, pp. 1300–1305.

[32] Y. Hao, C.-W. Ngo, and B. Huet, "Neighbourhood structure preserving cross-modal embedding for video hyperlinking," *IEEE Trans. Multimedia*, vol. 22, no. 1, pp. 188–200, 2019.

[33] J. Ji, C. Xu, X. Zhang, B. Wang, and X. Song, "Spatio-temporal memory attention for image captioning," *IEEE Trans. Image Process.*, vol. 29, pp. 7615–7628, 2020.

[34] Z. Niu and B. Mak, "Stochastic fine-grained labeling of multi-state sign glosses for continuous sign language recognition," in *European Conf. Computer Vision (ECCV)*, 2020, pp. 172–186.

[35] D. Li, C. Xu, X. Yu, K. Zhang, B. Swift, H. Suominen, and H. Li, "Tspnet: Hierarchical feature learning via temporal semantic pyramid for sign language translation," in *Annual Conf. Neural Information Processing Systems (NeurIPS)*, 2020.

[36] C.-C. Wang, C.-T. Chiu, C.-T. Huang, Y.-C. Ding, and L.-W. Wang, "Fast and accurate embedded dcnn for rgb-d based sign language recognition," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 1568–1572.

[37] H. Wang, C. Xiujuan, and C. Xilin, "A novel sign language recognition framework using hierarchical grassmann covariance matrix," *IEEE Trans. Multimedia*, vol. 21, no. 11, pp. 2806–2814, 2019.

[38] L.-C. Wang, R. Wang, D.-H. Kong, and B.-C. Yin, "Similarity assessment model for chinese sign language videos," *IEEE Trans. Multimedia*, vol. 16, no. 3, pp. 751–761, 2014.

[39] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "Learning clip representations for skeleton-based 3D action recognition," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2842–2855, 2018.

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

[41] L. Zhang, Y. Zhang, and X. Zheng, "Wisign: Ubiquitous american sign language recognition using commercial wi-fi devices," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 3, pp. 1–24, 2020.

[42] N. C. Camgoz, S. Hadfield, O. Koller, H. Ney, and R. Bowden, "Neural sign language translation," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 7784–7793.

[43] J.-M. Pérez-Rúa, V. Vielzeuf, S. Pateux, M. Baccouche, and F. Jurie, "MFAS: Multimodal fusion architecture search," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 6966–6975.

[44] N. C. Camgoz, O. Koller, S. Hadfield, and R. Bowden, "Multi-channel transformers for multi-articulatory sign language translation," in *European Conf. Computer Vision (ECCV)*, 2020, pp. 301–319.

[45] K. Yin and J. Read, "Better sign language translation with stmc-transformer," in *Int. Conf. Computational Linguistics (COLING)*, 2020, pp. 5975–5989.

[46] Z. Yu, J. Yu, J. Fan, and D. Tao, "Multi-modal factorized bilinear pooling with co-attention learning for visual question answering," in *IEEE Int. Conf. Computer Vision (ICCV)*, 2017, pp. 1821–1830.

[47] A. Zadeh, M. Chen, S. Poria, E. Cambria, and L.-P. Morency, "Tensor fusion network for multimodal sentiment analysis," in *Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2017, pp. 1103–1114.

[48] Z. Liu, Y. Shen, V. B. Lakshminarasimhan, P. P. Liang *et al.*, "Efficient low-rank multimodal fusion with modality-specific factors," in *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018, pp. 2247–2256.

[49] X. Liang, X. Shen, J. Feng, L. Lin, and S. Yan, "Semantic object parsing with graph lstm," in *European Conf. Computer Vision (ECCV)*, 2016, pp. 125–143.

[50] D. Beck, G. Haffari, and T. Cohn, "Graph-to-sequence learning using gated graph neural networks," in *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018, pp. 273–283.

[51] W. Fan, Y. Ma, Q. Li, Y. He, E. Zhao, J. Tang, and D. Yin, "Graph neural networks for social recommendation," in *Int. World Wide Web Conf. (WWW)*, 2019, pp. 417–426.

[52] M. Li, S. Chen, Y. Zhao, Y. Zhang, Y. Wang, and Q. Tian, "Dynamic multiscale graph neural networks for 3d skeleton based human motion prediction," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 214–223.

[53] C. C. de Amorim, D. Macêdo, and C. Zanchettin, "Spatial-temporal graph convolutional networks for sign language recognition," in *Int. Conf. Artificial Neural Networks (ICANN)*, 2019, pp. 646–657.

[54] A. Tunga, S. V. Nuthalapati, and J. Wachs, "Pose-based sign language recognition using gcn and bert," in *IEEE/CVF Winter Conf. Applications of Computer Vision (WACV)*, 2020, pp. 31–40.

[55] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Int. Conf. Learning Representations (ICLR)*, 2017.

[56] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Int. Conf. Machine Learning (ICML)*, 2006, pp. 369–376.

[57] P. Dreuw, C. Neidle, V. Athitsos, S. Sclaroff, and H. Ney, "Benchmark databases for video-based automatic sign language recognition." in *Int. Conf. Language Resources and Evaluation (LREC)*, 2008.

[58] X. Hua and L. Wang, "Sentence-level content planning and style specification for neural text generation," in *Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2019, pp. 591–602.

[59] Y. Yin, F. Meng, J. Su, C. Zhou, Z. Yang, J. Zhou, and J. Luo, "A novel graph-based multi-modal fusion encoder for neural machine translation," in *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020, pp. 3025–3035.

[60] L. Huang, W. Wang, J. Chen, and X.-Y. Wei, "Attention on attention for image captioning," in *IEEE Int. Conf. Computer Vision (ICCV)*, 2019, pp. 4634–4643.

[61] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2009.

[62] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Int. Conf. Machine Learning (ICML)*, 2015, pp. 448–456.

[63] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2015.

[64] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[65] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, "Sequence to sequence-video to text," in *IEEE Int. Conf. Computer Vision (ICCV)*, 2015, pp. 4534–4542.

[66] P. Pan, Z. Xu, Y. Yang, F. Wu, and Y. Zhuang, "Hierarchical recurrent neural encoder for video representation with application to captioning," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1029–1038.

[67] C. Wei, W. Zhou, J. Pu, and H. Li, "Deep grammatical multi-classifier for continuous sign language recognition," in *Int. Conf. Multimedia Big Data (BigMM)*, 2019, pp. 435–442.

[68] Y. Du, Y. Fu, and L. Wang, "Representation learning of temporal dynamics for skeleton-based action recognition," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3010–3022, 2016.

[69] X. Chen and M. Koskela, "Using appearance-based hand features for dynamic rgb-d gesture recognition," in *Int. Conf. Pattern Recognition (ICPR)*, 2014, pp. 411–416.

[70] P. Song, D. Guo, H. Xin, and M. Wang, "Parallel temporal encoder for sign language translation," in *IEEE Int. Conf. Image Processing (ICIP)*, 2019, pp. 1915–1919.

[71] P. Dreuw, D. Rybach, T. Deselaers, M. Zahedi, and H. Ney, "Speech recognition techniques for a sign language recognition system," in *Annual Conf. Int. Speech Communication Association (INTERSPEECH)*, 2007, pp. 2513–2516.

[72] P. Dreuw, J. Forster, T. Deselaers, and H. Ney, "Efficient approximations to model-based joint tracking and recognition of continuous sign language," in *Int. Conf. Automatic Face and Gesture Recognition (FG)*, 2008, pp. 1–6.

[73] P. Dreuw and H. Ney, "Visual modeling and feature adaptation in sign language recognition," in *ITG Conf. Voice Communication*, 2008, pp. 1–4.

[74] P. Dreuw, P. Steingrube, T. Deselaers, and H. Ney, "Smoothed disparity maps for continuous american sign language recognition," in *Iberian Conf. Pattern Recognition and Image Analysis (IbPRIA)*, 2009, pp. 24–31.

Shengeng Tang received the B.E. degree in computer science and technology from Hunan Normal University, China, in 2017. He is currently pursuing the Ph.D. degree at the School of Computer Science and Information Engineering, Hefei University of Technology, China. His research interests include multimedia content analysis and computer vision.

Dan Guo received the B.E. degree in computer science and technology from Yangtze University, China, in 2004, and the Ph.D. degree in system analysis and integration from Huazhong University of Science and Technology, China, in 2010. She is currently a professor at the School of Computer Science and Information Engineering, Hefei University of Technology, China. Her research interests include computer vision, machine learning, and intelligent multimedia content analysis.

Richang Hong received the Ph.D. degree from the University of Science and Technology of China, Hefei, China, in 2008. He is currently a professor at Hefei University of Technology, Hefei. His research interests include multimedia content analysis and social media, in which he has co-authored over 100 publications. He is a member of the ACM and an Executive Committee Member of the ACM SIGMM China Chapter. He was a recipient of the Best Paper Award at the ACM Multimedia 2010, the Best Paper Award at the ACM ICMR 2015, and the Honorable Mention of the IEEE TRANSACTIONS ON MULTIMEDIA Best Paper Award 2015. He was an Associate Editor of the IEEE Multimedia Magazine, Information Sciences and Signal Processing, Elsevier, and served as the Technical Program Chair of the MMM 2016, ICIMCS 2017, and PCM 2018.

Meng Wang (M09-SM17-F21) is a professor at Hefei University of Technology, China. He received his B.E. degree and Ph.D. degree in the Special Class for the Gifted Young and the Department of Electronic Engineering and Information Science from the University of Science and Technology of China (USTC), Hefei, China, in 2003 and 2008, respectively. His current research interests include multimedia content analysis, computer vision, and pattern recognition. He has authored over 200 book chapters, journal and conference papers in these areas. He is the recipient of the ACM SIGMM Rising Star Award 2014. He is an associate editor of IEEE Transactions on Knowledge and Data Engineering (IEEE TKDE), IEEE Transactions on Circuits and Systems for Video Technology (IEEE TCSVT), IEEE Transactions on Multimedia (IEEE TMM), and IEEE Transactions on Neural Networks and Learning Systems (IEEE TNNLS).