

# CVT-SLR: Contrastive Visual-Textual Transformation for Sign Language Recognition with Variational Alignment

Jiangbin Zheng<sup>1</sup>, Yile Wang<sup>1,2</sup>, Cheng Tan<sup>1</sup>, Siyuan Li<sup>1</sup>,  
Ge Wang<sup>1</sup>, Jun Xia<sup>1</sup>, Yidong Chen<sup>3</sup>, Stan Z. Li<sup>1\*</sup>

<sup>1</sup>AI Lab, Research Center for Industries of the Future, Westlake University

<sup>2</sup>Institute for AI Industry Research (AIR), Tsinghua University

<sup>3</sup>School of Informatics, Xiamen University

{zhengjiangbin, wangyile, tancheng, lisiyuan, wangge, xiajun, Stan.ZQ.Li}@westlake.edu.cn  
ydchen@xmu.edu.cn

## Abstract

*Sign language recognition (SLR) is a weakly supervised task that annotates sign videos as textual glosses. Recent studies show that insufficient training caused by the lack of large-scale available sign datasets becomes the main bottleneck for SLR. Most SLR works thereby adopt pre-trained visual modules and develop two mainstream solutions. The multi-stream architectures extend multi-cue visual features, yielding the current SOTA performances but requiring complex designs and might introduce potential noise. Alternatively, the advanced single-cue SLR frameworks using explicit cross-modal alignment between visual and textual modalities are simple and effective, potentially competitive with the multi-cue framework. In this work, we propose a novel contrastive visual-textual transformation for SLR, CVT-SLR, to fully explore the pre-trained knowledge of both the visual and language modalities. Based on the single-cue cross-modal alignment framework, we propose a variational autoencoder (VAE) for pre-trained contextual knowledge while introducing the complete pretrained language module. The VAE implicitly aligns visual and textual modalities while benefiting from pretrained contextual knowledge as the traditional contextual module. Meanwhile, a contrastive cross-modal alignment algorithm is designed to explicitly enhance the consistency constraints. Extensive experiments on public datasets (PHOENIX-2014 and PHOENIX-2014T) demonstrate that our proposed CVT-SLR consistently outperforms existing single-cue methods and even outperforms SOTA multi-cue methods. The source codes and models are available at <https://github.com/binbinjiang/CVT-SLR>.*

\*Corresponding author.

## 1. Introduction

As a special visual natural language, sign language is the primary communication medium of the deaf community [19]. With the progress of deep learning [1, 17, 25, 39, 42], sign language recognition (SLR) has emerged as a multi-modal task that aims to annotate sign videos into textual sign glosses. However, a significant dilemma of SLR is the lack of publicly available sign language datasets. For example, the most commonly-used PHOENIX-2014 [23] and PHOENIX-2014T [2] datasets only include about 10K pairs of sign videos and gloss annotations, which are far from training a robust SLR system with full supervision as typical vision-language cross-modal tasks [34]. Therefore, data limitation that may easily lead to insufficient training or overfitting problems is the main bottleneck of SLR tasks.

The development of weakly supervised SLR has witnessed most of the improvement efforts focus on the visual module (e.g., CNN) [9, 10, 15, 29, 32, 33]. Transferring pre-trained visual networks from general domains of human actions becomes a consensus to alleviate the low-resource limitation. The mainstream multi-stream SLR framework extends the pretrained visual module with multi-cue visual information [3, 22, 24, 43, 48, 50], including global features and regional features such as hands and faces in independent streams. The theoretical support for this approach comes from sign language linguistics, where sign language utilizes multiple complementary channels (e.g., hand shapes, facial expressions) to convey information [3]. The multi-cue mechanism essentially exploits hard attention to key information, yielding the current SOTA performances. However, the multi-cue framework is complex (e.g., cropping multiple regions, requiring more parameters), and the fusion of multiple streams might introduce additional potential noise.

Another mainstream advanced solution is the single-cue

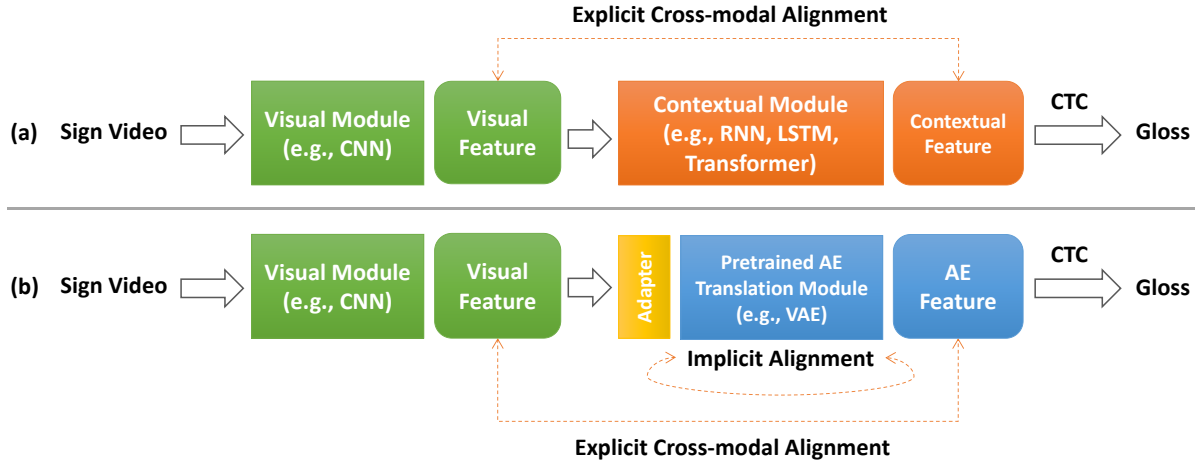


Figure 1. (a) An advanced single-cue SLR framework with explicit cross-modal alignment; (b) Our proposed single-cue SLR framework with explicit cross-modal alignment and implicit autoencoder alignment. Both frameworks use pretrained visual features. But our framework uses the autoencoder module to replace the mainstream contextual module, which not only includes the functions of the contextual module but also can introduce complete pretrained language knowledge and implicit cross-modal alignment. To maximize the preservation of the complete pretrained language parameters and migrated visual features, a video-gloss adapter is introduced.

cross-modal alignment framework [15, 28], which consists of a pretrained visual module followed by a contextual module (e.g., RNN, LSTM, Transformer) and a Connectionist Temporal Classification (CTC) [14] based alignment module for gloss generation, as shown in Figure 1 (a). Explicit cross-modal alignment constraints further improve feature interactions [15, 28, 38], which could be treated as a kind of consistency between two different modalities [50] facilitating the visual module learn long-term temporal information from contextual module [13, 37]. The cross-modal alignment framework is simple and effective, potentially competitive with the multi-cue framework. Despite the advanced performance of complex multi-cue architectures with pretrained visual modules, the cross-modal consistency is a more elegant design for practical usage. It also implies the potential of prior contextual linguistic knowledge, which has been overlooked by existing SLR works.

In this work, we propose a novel contrastive visual-textual transformation framework for SLR, called CVT-SLR, to fully explore the pretrained knowledge of both the visual and language modalities, as shown in Figure 1 (b). Based on the single-cue cross-modal alignment framework, CVT-SLR keeps the pretrained visual module but replaces the traditional contextual module with a variational autoencoder (VAE). Since a full encoder-decoder architecture is used, the VAE is responsible for learning pretrained contextual knowledge based on a pseudo-translation task while introducing the complete pretrained language module. In addition, the VAE maintains the consistency of input and output modalities due to the form of an autoencoder, playing an implicit cross-modal alignment role. Furthermore, inspired by contrastive learning [4–6, 34], we introduce a contrastive

alignment algorithm that focuses on both positive and negative samples to enhance explicit cross-modal consistency constraints. Extensive quantitative experiments conducted on the public datasets PHOENIX-2014 and PHOENIX-2014T demonstrate the advance of the proposed CVT-SLR framework. Through ablation study and qualitative analysis, we also verify the effectiveness of introducing pretrained language knowledge and the new consistency constraint mechanism.

Our main contributions are as follows:

- A novel visual-textual transformation-based SLR framework is proposed, which introduces fully pretrained language knowledge for the first time and provides new approaches for other cross-modal tasks.
- New alignment methods are proposed for cross-modal consistency constraints: a) exploiting the special properties of the autoencoder to implicitly align visual and textual modalities; b) introducing an explicit contrastive cross-modal alignment method.
- The proposed single-cue SLR framework not only outperforms existing single-cue baselines by a large margin but even surpasses SOTA multi-cue baselines.

## 2. Related Work

SLR tasks are generally divided into three categories [19]: finger-spelling recognition [18, 30, 31], isolated word recognition [26, 40, 41, 49] and continuous sign sentence recognition [8, 11, 12, 29, 33]. In the early days, SLR works mainly focused on lexical-level tasks, such as finger-spelling recognition and isolated word recognition. Nowadays, the more practical continuous sign sentence recogni-

tion task has become mainstream sign language research. In this work, the mentioned SLRs refer specifically to continuous sign sentence recognition.

The recent SLR works [8, 11, 15, 29] are summarized based on these three aspects: feature extraction, recognition, and alignment. Mostly, the components of feature extraction are composed of the visual module and the contextual module. The visual module encodes short-term spatial information, while the contextual module encodes long-term context information. Based on the extracted features, the classifiers can get a posterior probability for recognition. And the alignment module is required to find the proper alignment between clips and glosses to ensure an accurate training procedure.

A few studies extend the common SLR framework by incorporating multi-cue information [22, 24, 43, 44, 48]. The multi-cue features mainly include information from hand shapes, facial expressions, mouths, and poses. The current SOTA SLR works are based on multi-cue mechanisms, such as C<sup>2</sup>SLR [50]. Recently, some works show that explicitly enforcing the consistency between the visual and textual modules can also get comparable performance, although only single-cue features are considered. For example, VAC [28] treats the visual and textual modules as the student and teacher, respectively, and achieves knowledge distillation to align the visual and the textual modalities. Similarly, SMKD [15] achieves knowledge transfer by further sharing classifiers between different modalities. Note that SMKD is the current SOTA single-cue method.

It is worth mentioning that neural sign language translation [2, 3, 7, 46] is another popular sign language task. As a distinction, the neural sign language translation task aims to convert sign videos into equivalent spoken language translations, further focusing on semantic relationships and adjusting word order [2], while the SLR task only annotates sign videos into gloss sequences in the frame order.

### 3. Methods

As shown in Figure 2, our training pipeline is roughly divided into two steps. Step 1 is to pretrain a VAE network as the textual module. And step 2 is to transfer the off-the-shelf visual module (usually a publicly available CNN trained on Kinetics/ImageNet) and the pretrained textual module from step 1 into the CVT-SLR framework. A video-gloss adapter is used to bridge these two pretrained modules.

#### 3.1. Textual Model Pretraining

Our VAE model is an asymmetric encoder-decoder architecture, where the encoder consists of self-attention layers, and the decoder consists of Bi-LSTM layers, as shown in the top part of Figure 2. To pretrain the VAE for our proposed SLR framework solely based on the gloss data from the PHOENIX-14 training set (excluding the development

set and testing set), we construct a pseudo-translation task, Gloss2Gloss, which enables the VAE to learn the contextual semantic knowledge. Formally, it is assumed that the input gloss of the encoder is denoted as  $G$ , and the generative gloss of the decoder is denoted as  $G^*$ .

**Posterior Self-attention Encoder.** We employ the neural approximation, and reparameterization [21, 35] techniques to approximate the posterior self-attention encoder for generating the posterior distribution  $q_\phi(z|G)$ , where  $\phi$  denotes the variational parameters of the encoder. The  $q_\phi(z|G)$  is a diagonal Gaussian distribution  $N(\mu, \text{diag}(\sigma^2))$ , and two single-layer MLPs are used to parameterize the mean  $\mu$  and the variance  $\sigma^2$ , respectively, as:

$$z = \mu + \sigma \odot \epsilon, \quad (1)$$

where  $\epsilon$  is a standard Gaussian noise, and  $\odot$  denotes an element-wise product.

**Generative Bi-LSTM Decoder.** Given the continuous latent variable  $z$ , then the translation target  $G^*$  is generated from  $z$  as:

$$p_\theta(G^*, z) = p_\theta(G^*|z)p_\theta(z), \quad (2)$$

where  $\theta$  denotes the parameters of the generative Bi-LSTM decoder, and  $p_\theta(G^*|z)$  is the conditional distribution that models the generation procedure estimated via the decoder.

**Variational Optimization.** Both the generative model  $p_\theta(G^*, z)$  and the posterior inference model  $q_\phi(z|G)$  are incorporated tightly in an end-to-end manner, and we can apply a standard back-propagation to optimize the gradient of the variational lower bound:

$$\begin{aligned} \mathcal{L}_{\text{VAE}}(\theta, \phi; G) = & -\text{KL}(q_\phi(z|G)||p_\theta(z)) \\ & + \mathbb{E}_{q_\phi(z|G)}[\log p_\theta(G^*|z)] \leq \log p_\theta(G^*), \end{aligned} \quad (3)$$

where KL denotes the Kullback-Leibler divergence and the prior  $p_\theta(z)$  denotes a standard Gaussian distribution. Practically, the first KL term of Eq. (3) can be written as:

$$\mathcal{L}_{\text{KL}} = -\frac{1}{2}(\log \sigma^2 - \sigma^2 - \mu^2 + 1). \quad (4)$$

Since our main target is a Gloss2Gloss translation task, we employ a cross-entropy loss to maximize the generative gloss probability as the second term of Eq. (3):

$$\mathcal{L}_{\text{gloss2gloss}} = \text{CrossEntropy}(G, G^*). \quad (5)$$

Therefore, the overall optimization objective of textual model pretraining is a joint loss of  $\mathcal{L}_{\text{KL}}$  and  $\mathcal{L}_{\text{gloss2gloss}}$  as:

$$\mathcal{L}_{\text{VAE}} = \mathcal{L}_{\text{KL}} + \mathcal{L}_{\text{gloss2gloss}}. \quad (6)$$

#### 3.2. Training CVT-SLR Framework

##### 3.2.1 Training Pipeline

In the CVT-SLR framework, we sequentially concatenate the pretrained visual CNN module and the pretrained textual VAE module through a video-gloss adapter, as shown

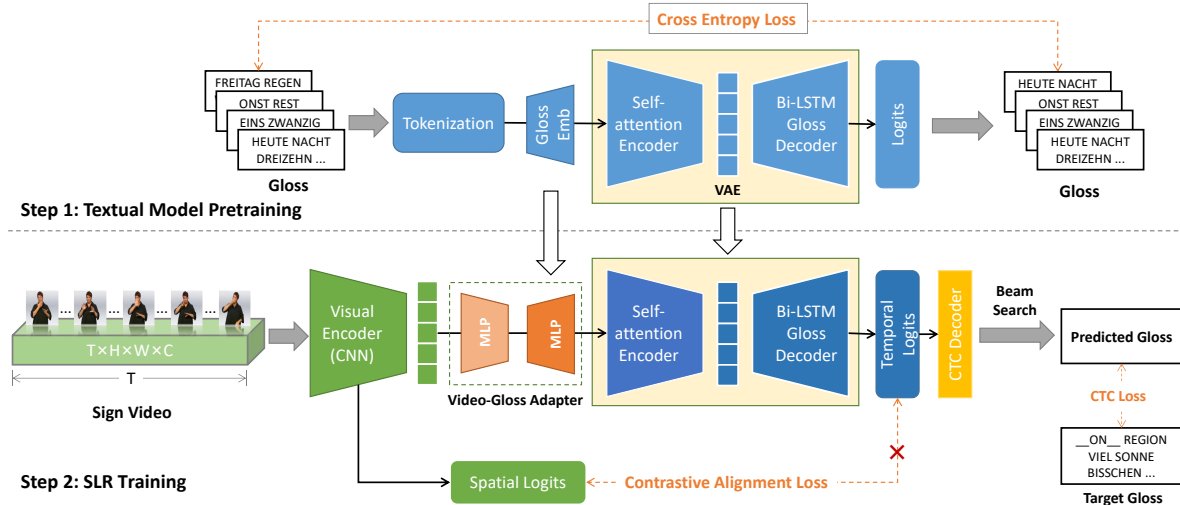


Figure 2. Overview of proposed CVT-SLR framework. Our training pipeline is divided into two steps. Step 1: Pretrain the VAE module with the self-supervised gloss-to-gloss pseudo-translation task. Step 2: Train the CVT-SLR based on the VAE model from Step 1.

in Figure 2. The pretrained ResNet18 model [17] is selected as the frame-wise feature extractor, which is trained on large-scale human action datasets. For the VAE module, although Gaussian noise is introduced in the reparameterization process in step 1, we do not add any noise after migrating the VAE to the SLR framework in step 2. In this way, the reparameterization module of the VAE degenerates into two common linear layers, i.e.,  $\epsilon$  is removed in Eq. (1).

Formally, given a sign video with  $T$  RGB frames  $\mathcal{F} = \{\mathcal{F}_t\}_{t=1}^T \in \mathbb{R}^{T \times H \times W \times 3}$ , the visual module, which is composed of a stack of 2D-CNN layers and a global average pooling layer, first extracts spatial features  $v = \{v_t\}_{t=1}^T \in \mathbb{R}^{T \times d}$ . Then the textual module further extracts temporal features  $s = \{s_t\}_{t=1}^T \in \mathbb{R}^{T \times d}$ . Finally, the alignment module utilizes CTC to compute the conditional probability of the gloss label sequence  $p(\mathcal{G}|\mathcal{F})$  based on temporal features  $s$ , where  $\mathcal{G} = \{\mathcal{G}_i\}_{i=1}^N$  is a generated sign gloss sequence with  $N$  glosses. A shared classifier is used between the visual module and the textual module to generate spatial and temporal probability, respectively. And the proposed contrastive cross-modal alignment algorithm further enhances the consistency across different modalities.

### 3.2.2 Video-Gloss Adapter

To maximize the preservation of both pretrained visual and textual knowledge, we use a video-gloss adapter to connect spatial-temporal features, as shown in Figure 2. The video-gloss adapter is simply implemented as a fully-connected MLP with two hidden layers. We take the CNN output probabilities as the input of the adapter where the final fully connected layer is initialized using the weights of the pretrained word embedding from the pretrained VAE [7].

Since both the input and output are in the same textual

form during the textual model pretraining in step 1, the VAE plays an implicit cross-modal alignment role after being transferred to the CVT-SLR framework in step 2, which is close to the effect of consistency. Therefore, the video-gloss adapter is essentially a mediator for activating the consistency between the spatial and temporal modalities.

### 3.2.3 CTC Alignment

Due to the spatial-temporal nature of sign videos, glosses have a one-to-many mapping to video frames but share the same ordering. Unlike spoken language texts, the continuous gloss annotations are chronologically consistent with the sign videos. Concretely, for an input sign video  $\mathcal{F}$  and the corresponding generated gloss sequence  $\mathcal{G}$ , we use CTC to compute  $p(\mathcal{G}|\mathcal{F})$  by marginalizing over all possible  $\mathcal{F}$  to  $\mathcal{G}$  alignments:

$$p(\mathcal{G}|\mathcal{F}) = \sum_{\pi \in \mathcal{B}} p(\pi|\mathcal{F}), \quad (7)$$

where  $\pi$  denotes a path and  $\mathcal{B}$  is the set of all viable paths corresponding to  $\mathcal{G}$ . The probability  $p(\pi|\mathcal{F})$  is computed by the textual module. The CTC loss is then formulated as:

$$\mathcal{L}_{\text{ctc}} = -\log p(\mathcal{G}^*|\mathcal{F}), \quad (8)$$

where  $\mathcal{G}^*$  is the ground truth gloss sequence.

### 3.2.4 Contrastive Alignment

Aligning both visual and textual modalities help improve SLR [28]. To enforce alignment constraints across modalities, we propose a contrastive alignment loss. In addition, shared classifiers are also used as an improvement technique between visual and textual features [15].

Groups	Models	Dev (%)		Test (%)		Cues
		DEL/INS	WER	DEL/INS	WER	
Group 1	SubUNet [9]	14.6/4.0	40.8	14.3/4.0	40.7	video
	Staged-Opt [9]	13.7/7.3	39.4	12.2/7.5	38.7	video
	Align-iOpt [33]	12.6/2.6	37.1	13.0/2.5	36.7	video
	DPD+TEM [47]	9.5/3.2	35.6	9.3/3.1	34.5	video
	Re-Sign [24]	-	27.1	-	26.8	video
	SFL [29]	7.9/6.5	26.2	7.5/6.3	26.8	video
	DNF [11]	7.8/3.5	23.8	7.8/3.4	24.4	video
	FCN [8]	-	23.7	-	23.9	video
	VAC [28]	7.9/2.5	21.2	8.4/2.6	22.3	video
	CMA [32]	7.3/2.7	21.3	7.3/2.4	21.9	video
	SFL [29]	10.3/4.1	24.9	10.4/3.6	25.3	video
	VL-SLT [7]	-	21.9	-	22.5	video
	SMKD [15]	6.8/2.5	<u>20.8</u>	6.3/2.3	<u>21.0</u>	video
Group 2	DNF [11]	7.3/3.3	23.1	6.7/3.3	22.9	video+optical flow
	STMC [48]	7.7/3.4	21.1	7.4/2.6	20.7	video+hand+face+pose
	C <sup>2</sup> SLR [50]	-	<u>20.5</u>	-	<u>20.4</u>	video+keypoints
Group 3	Ours <sub>1</sub> (w/o VAE+Contra)	7.1/3.0	21.1	7.3/2.9	21.4	video
	Ours <sub>2</sub> (w/ VAE)	6.5/2.4	20.2	6.3/2.2	20.3	video
	Ours <sub>3</sub> (w/ Contra)	6.7/2.7	20.4	6.4/2.5	20.7	video
	Ours <sub>4</sub> (w/ VAE+Contra)	6.4/2.6	<b>19.8</b>	6.1/2.3	<b>20.1</b>	video

Table 1. Performance comparison (%) on PHOENIX-14 dataset. DEL/INS: deletion error and insertion error. The best results and SOTA baseline for each group are marked as **bold** and underlined.

In previous SLR works, cross-modal alignment only focuses on positive samples [15, 48]. Inspired by contrastive learning [5, 16, 45], we construct both positive and negative samples in the same mini-batch and implement a contrastive cross-modal alignment method to ensure that similar features are closer while different features are farther apart. Given that the normalized spatial features from CNN as  $\mathcal{S}_{\text{logits}} \in \mathbb{R}^{B \times T \times d}$ , and the normalized temporal features from VAE as  $\mathcal{V}_{\text{logits}} \in \mathbb{R}^{B \times T \times d}$ , where  $B$  denotes the number of samples. Then we compute pair matrices firstly as:

$$\mathcal{S}2\mathcal{V}_{\text{pair}} = \mathcal{S}_{\text{logits}} \times \mathcal{V}_{\text{logits}}^T, \quad (9)$$

$$\mathcal{V}2\mathcal{S}_{\text{pair}} = \mathcal{V}_{\text{logits}} \times \mathcal{S}_{\text{logits}}^T, \quad (10)$$

Taking  $\mathcal{S}2\mathcal{V}_{\text{pair}} \in \mathbb{R}^{B \times B}$  pair matrix (denoted as  $\mathcal{P}$ ) as an example,  $\mathcal{P}[i, j] (0 \leq i, j < B)$  represents the similarity value between the visual feature of the  $i$ -th batch and the textual feature of  $j$ -th batch. The visual and textual features from the same input instances are positive samples, and the features from other different input instances are negative samples. Hence, in the matrix  $\mathcal{P}$ , the diagonal similarity values are from positive sample pairs, and the rest values are from negative sample pairs. To make the loss computation differentiable and easy, we convert the computation of positive and negative pairs into a binary classification task [34]. The value  $i$  corresponding to the  $i$ -th row in the matrix  $\mathcal{P}$

is a positive label. Therefore, the corresponding labels of  $\mathcal{P}$  are Labels =  $\{0, 1, \dots, i, \dots, B\}$ , then the contrastive alignment loss related to  $\mathcal{S}2\mathcal{V}_{\text{pair}}$  as:

$$\mathcal{L}_{\text{align}_{\mathcal{S}}} = \text{CrossEntropy}(\text{Softmax}(\mathcal{S}2\mathcal{V}_{\text{pair}}), \text{Labels}). \quad (11)$$

Similarly, the  $\mathcal{V}2\mathcal{S}_{\text{pair}}$  temporal-to-spatial alignment loss as:

$$\mathcal{L}_{\text{align}_{\mathcal{V}}} = \text{CrossEntropy}(\text{Softmax}(\mathcal{V}2\mathcal{S}_{\text{pair}}), \text{Labels}). \quad (12)$$

Therefore, the complete contrastive alignment loss as:

$$\mathcal{L}_{\text{align}} = \frac{1}{2}(\mathcal{L}_{\text{align}_{\mathcal{S}}} + \mathcal{L}_{\text{align}_{\mathcal{V}}}). \quad (13)$$

Overall, the objective of CVT-SLR is jointly optimized CTC loss  $\mathcal{L}_{\text{ctc}}$  and contrastive alignment loss  $\mathcal{L}_{\text{align}}$  as:

$$\mathcal{L}_{\text{CVT-SLR}} = \mathcal{L}_{\text{ctc}} + \mathcal{L}_{\text{align}}. \quad (14)$$

## 4. Experiments

### 4.1. Settings

**Datasets.** We use PHOENIX-2014 [23] and PHOENIX-2014T [2] as datasets, where RGB videos and their corresponding annotations are provided. PHOENIX-2014 is a German SLR dataset about weather forecasts with a vocabulary size of 1,081, which is divided into three parts: 5,672



Groups	Models	WER		Cues
		Dev(%)	Test(%)	
Group 1	SFL [29]	25.1	26.1	video
	CNN+LSTM+HMM [22]	24.5	26.5	video
	SLT [3]	24.9	24.6	video
	FCN [8]	23.3	25.1	video
	SMKD [15]	<u>20.8</u>	<u>22.4</u>	video
Group 2	CNN+LSTM+HMM [22]	24.5	25.4	video+mouth
	CNN+LSTM+HMM [22]	22.1	24.1	video+mouth+hand
	SLT [3]	24.6	24.5	video+text
	STMC [48]	<u>19.6</u>	21.0	video+hand+face+pose
	C <sup>2</sup> SLR [50]	20.2	<u>20.4</u>	video+keypoints
Group 3	Ours <sub>1</sub> (w/o VAE+Contra)	21.8	22.0	video
	Ours <sub>2</sub> (w/ VAE)	20.1	20.4	video
	Ours <sub>3</sub> (w/ Contra)	21.0	21.5	video
	Ours <sub>4</sub> (w/ VAE+Contra)	<b>19.4</b>	<b>20.3</b>	video

Table 2. Performance comparison (%) on PHOENIX14T dataset. The best results and SOTA baseline for each group are marked as **bold** and underlined, respectively.

instances for training, 540 for validation, and 629 for testing, respectively. And PHOENIX-2014T is an extension of PHOENIX-2014 with a vocabulary size of 1,085, which is also divided into three parts: 7,096 instances for training, 519 for validation, and 642 for testing, respectively.

**Evaluation Metric.** Word error rate (WER) is the most widely-used metric to evaluate the SLR performance, which measures the number of necessary insertions, substitutions, and deletions in the recognized sentences when compared to the reference sentences. WER is an edit distance, indicating the least number of operations of substitutions (#sub), insertions (#ins), and deletions (#del) to transform the predicted sentences into the reference sequences (#reference):

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

**Implementation Details.** The videos are resized to  $256 \times 256$  and then cropped to  $224 \times 224$ . During SLR training, we use random crop and horizontal flips (50%) for data augmentation. During SLR testing, we only adopt center cropping. Beam Search is used for gloss generation. Our visual module has the same configuration as [15]. The VAE module contains two multi-head self-attention layers with 4 heads and two Bi-LSTM layers with  $2 \times 512$  dimensional hidden states sequentially. Adam [20] optimizer is used to pretrain our VAE network with a batch size of 16, an initial learning rate of  $1e-4$ . And AdamW [27] optimizer is used to train our SLR framework with a batch size of 8, a learning rate of  $1e-4$ , and a weight decay of  $1e-4$ . And 1 NVIDIA A100 80GB GPU is used.

## 4.2. Main Results

Table 1 and Table 2 show our comparisons with two-group different baselines, where Group 1 includes

single-cue baselines and Group 2 includes multi-cue baselines. Ours<sub>number</sub> denote our trained CVT-SLR framework, which have only two variables, “VAE” and “Contra”. Note that “w/o (w/) VAE” denotes the VAE module in CVT-SLR framework without (with) transferred pretrained parameters, and “w/o (w/) Contra” denotes training CVT-SLR framework without (with) contrastive alignment loss.

**Evaluation on PHOENIX-14.** Table 1 shows our methods compared to other methods on PHOENIX-14. The optimal WER scores of our single-cue CVT-SLR (Ours<sub>4</sub>) on the development set and testing set are 19.8% and 20.1%, respectively. And Ours<sub>4</sub> achieves the best performance among video-based only/single-cue methods in Group 1 (Ours<sub>4</sub> vs. SMKD). Furthermore, while no extra cues are used, our CVT-SLR achieves the best result among the models trained with extra cues (i.e., multi-cue methods) in Group 2 (Ours<sub>4</sub> vs. C<sup>2</sup>SLR).

From within-group comparisons in Group 3, we find that the case, Ours<sub>1</sub>, which neither migrates the pretrained VAE parameters nor uses the contrastive alignment constraint, performs the worst. Nevertheless, relative to other groups, it still performs well. Based on Ours<sub>1</sub>, by only configuring the pretrained VAE parameters (Ours<sub>2</sub>) or the contrastive alignment constraints (Ours<sub>3</sub>), the CVT-SLR framework can also gain gains, and the pretrained VAE brings more obvious gains (+0.9%/+1.1% vs. +0.7%/+0.7%). Overall, Group 3 demonstrates that both the prior language knowledge from the pretrained textual module and the contrastive constraints through cross-modal alignment play important enhancement roles for our SLR framework.

**Evaluation on PHOENIX-14T.** In Table 2, we evaluate the CVT-SLR on PHOENIX-14T. We can see that our optimal model (Ours<sub>4</sub>) in this dataset also achieves the best per-

#	Pre-Visual	Pre-Textual	WER	
			Dev(%)	Test(%)
1	✗	✗	89.2	88.5
2	✗	✓	30.8	31.4
3	✓	✗	20.4	20.7
4	✓	✓	<b>19.8</b>	<b>20.1</b>

Table 3. Ablation study of pretrained modules on PHOENIX-14 (Pre-Visual: pretrained visual module, Pre-Language: pretrained textual module). The best results are **bolded**.

#	CTC Weight	Align Weight	WER	
			Dev(%)	Test(%)
1	1.0	0	20.2	20.3
2	1.0	1.0	20.0	20.2
3	1.0	10.0	<b>19.8</b>	<b>20.1</b>
4	1.0	100.0	20.1	20.6
5	1.0	1000.0	20.5	20.9

Table 4. Ablation study of varied weights of the CTC loss and the alignment loss on PHOENIX-14. The best results are **bolded**.

formance (development set: 19.4% and testing set: 20.3%) with video-based only/single-cue information compared to Group 1 (Ours<sub>4</sub> vs. SMKD). It is worth mentioning that the SOTA single-cue method (SMKD) fails to surpass the multi-cue baselines in their work on PHOENIX-14T. And our CVT-SLR makes up for their shortcomings while outperforming the SOTA multi-cue baselines in Group 2 on this dataset (Ours<sub>4</sub> vs. C<sup>2</sup>SLR).

Compared within Group 3 on the PHOENIX-14T, the observed conclusions are more or less the same as on the PHOENIX-14, i.e., both the pretrained textual module and the proposed contrastive alignment constraint have a beneficial effect on our CVT-SLR framework. But the pretrained textual module gain on PHOENIX-14T is more significant than that on PHOENIX-14 (+1.7%/+1.6% vs. +0.9%/+1.1%), indicating that prior language knowledge is more beneficial for this dataset.

### 4.3. Ablation Study

We study the effectiveness of the proposed pretrained modules and alignment loss weights in our framework.

#### 4.3.1 Pretrained Modules

To illustrate the importance of the pretrained modules in our CVT-SLR framework, we conduct the ablation studies of both the pretrained visual module (CNN module) and the pretrained textual module (VAE module) on PHOENIX-14, as verified in Table 3.

Overall, both pretrained modules are useful. Comparing #1 and #3, we find that pretrained visual features account for a larger determinant of performance, which is consistent

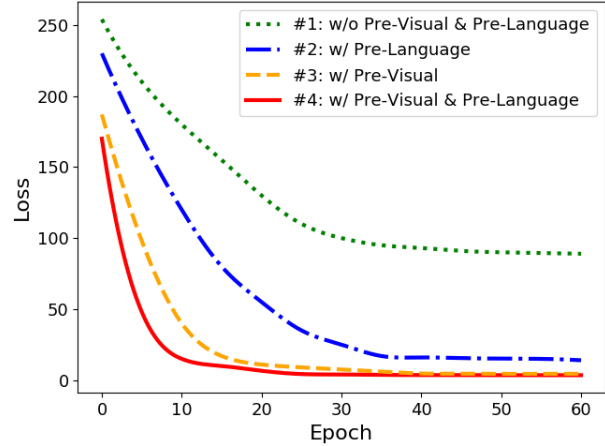


Figure 3. The training loss curves corresponding to the ablation study of pretrained modules in Table 3.

with our reason for retaining the pretrained visual knowledge. Surprisingly, comparing #1 and #2, we find that without pretrained parameters of the visual module, i.e., using only the pretrained textual module, our CVT-SLR performance is also greatly improved. Notably, this is a new observation that has not been explored in previous SLR works extensively. These comparisons demonstrate that the pretrained VAE module has learned sufficient language knowledge to improve the SLR task. Comparing #3 with #4 further confirms this point.

To verify the effectiveness of the pretrained modules for the training convergence speed, training loss curves corresponding to Table 3 are shown in Figure 3. Specifically, comparing #1 and #2 (or #3 and #4), configuring a pretrained textual module can promote faster convergence and reach a stable state earlier. Similarly, comparing #1 and #3 (or #2 and #4), a pretrained visual module can accelerate convergence speed compared to the pretrained textual module. As can be seen from the loss curves, the degrees of convergence are positively correlated with the final evaluation performances of WER scores.

#### 4.3.2 Alignment Loss Weights

Training CVT-SLR to recognize glosses is optimized jointly by CTC loss and contrastive alignment loss. To explore the effect of loss weights, we conduct an ablation study of varied weights of the CTC loss and the proposed contrastive alignment loss, as shown in Table 4. For simplicity, we set the weights of CTC loss to a fixed value of 1.0. In Table 4, #1, without contrastive alignment loss (weight=0), performs worse than the models with contrastive alignment in an optimal configuration (#2, #3), which also illustrates the gain effect of the proposed contrastive alignment. As shown in #2-#5 in Table 4, we keep increasing the weight of the alignment loss from 1.0 to 1000.0. As the weight of the contrastive loss increases, the performance starts to rise, reach-

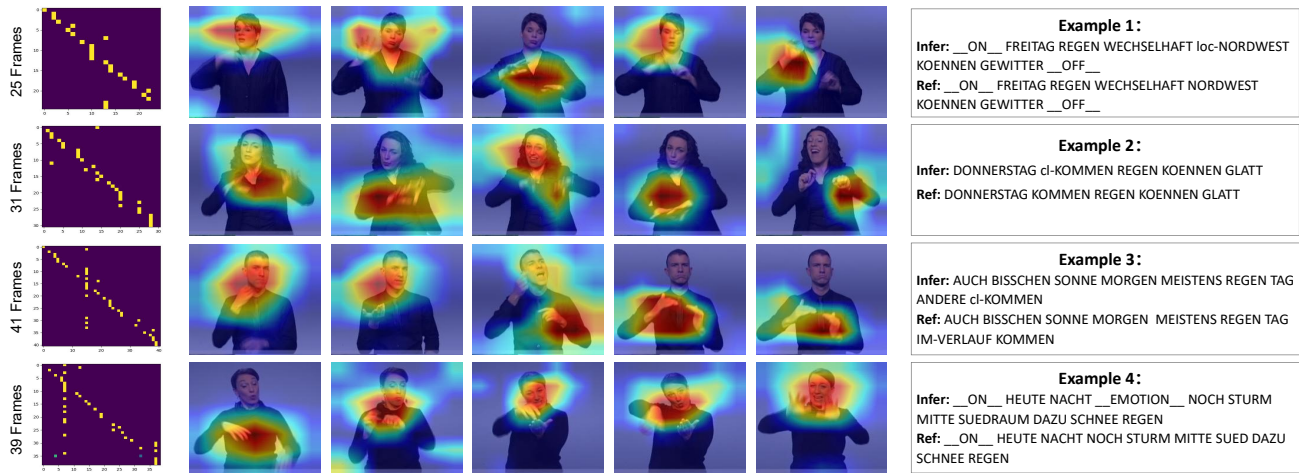


Figure 4. Four examples with cross-modal alignment matrices (left), saliency maps (middle), and generated glosses (right) on the PHOENIX-14 test set. Cross-modal alignment matrices show the alignment between visual and textual features, while saliency maps show the highest activation regions focused on gestures.

ing a peak when the weight value is 10.0. As the weight increases and the value is greater than 10.0, the performance starts to decline instead. This ablation study indicates the important role of the proposed contrastive alignment. By balancing the weights between CTC loss and contrastive alignment loss, the model can achieve optimal performance.

#### 4.4. Qualitative Analysis

We randomly select four examples from the test set of PHOENIX-14 for qualitative analysis. Each example displays an alignment matrix, a series of saliency maps, and generated glosses in Figure 4. An alignment matrix shows alignment relationships between visual and textual features, while a saliency map shows the highest activations.

##### Visualization on Cross-modal Alignment Matrices.

From these four alignment matrices in Figure 4, we can see that the reference and inferred glosses are almost the same with quite low WER scores, and the performances are relatively perfect. Hence, the highlighted areas of the alignment matrices are concentrated near the diagonal, which means that our contrastive alignment algorithm plays a crucial role in aligning visual and textual features.

**Visualization on Saliency Maps.** We visualize the key parts of the sign video frames in focus by using Grad-CAM [36] as shown in Figure 4. The results show that, for well-performing inferences, significant spatial features are mainly focused on the face and hand regions in the saliency maps. This observation is consistent with the sign linguistic knowledge [3, 19], where sign language conveys information mainly relying on hand shapes and facial expressions.

## 5. Conclusion

In this work, we propose a novel contrastive visual-textual transformation framework for SLR, called CVT-

SLR, which further alleviates insufficient training by fully exploring the pretrained knowledge of both the visual and language modalities. We introduce prior language knowledge into the single-cue CVT-SLR for the first time via a VAE-based pretrained textual module. Furthermore, we propose new methods for cross-modal consistency constraints. One method takes advantage of the properties of the autoencoder to implicitly align the visual and textual modalities. While another method introduces a contrastive cross-modal alignment for explicit consistency. To prove the proposed CVT-SLR framework, extensive quantitative experiments and qualitative analysis are conducted, showing that our single-cue CVT-SLR not only outperforms single-cue baselines by a large margin but also surpasses SOTA multi-cue methods.

Overall, we hope our CVT-SLR can inspire the SLR community to mine pretrained modules from new aspects and design a more efficient SLR framework for practical usage. In future work, we plan to introduce a large-scale pretrained language model to guide textual module learning and further explore the potential of prior contextual linguistic knowledge for SLR tasks.

## Acknowledgement

We thank the anonymous reviewers for their constructive and helpful reviews. This work was supported by the National Key R&D Program of China (Project No. 2022ZD0115100), the National Natural Science Foundation of China (Project No. U21A20427), the Competitive Research Fund (Project No. WU2022A009) from the Westlake Center for Synthetic Biology and Integrated Bioengineering.



## References

- [1] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. In *ICLR*, 2022. 1
- [2] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. Neural sign language translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7784–7793, 2018. 1, 3, 5
- [3] Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard 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*, pages 10023–10033, 2020. 1, 3, 6, 8
- [4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020. 2
- [5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 2, 5
- [6] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020. 2
- [7] Yutong Chen, Fangyun Wei, Xiao Sun, Zhirong Wu, and Stephen 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*, pages 5120–5130, 2022. 3, 4, 5
- [8] Ka Leong Cheng, Zhaoyang Yang, Qifeng Chen, and Yu-Wing Tai. Fully convolutional networks for continuous sign language recognition. In *European Conference on Computer Vision*, pages 697–714. Springer, 2020. 2, 3, 5, 6
- [9] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden. Subunets: End-to-end hand shape and continuous sign language recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 3056–3065, 2017. 1, 5
- [10] Runpeng Cui, Hu Liu, and Changshui Zhang. Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7361–7369, 2017. 1
- [11] Runpeng Cui, Hu Liu, and Changshui Zhang. A deep neural framework for continuous sign language recognition by iterative training. *IEEE Transactions on Multimedia*, 21(7):1880–1891, 2019. 2, 3, 5
- [12] Cleison Correia de Amorim, David Macêdo, and Cleber Zanchettin. Spatial-temporal graph convolutional networks for sign language recognition. In *International Conference on Artificial Neural Networks*, pages 646–657. Springer, 2019. 2
- [13] Zhangyang Gao, Cheng Tan, Lirong Wu, and Stan Z Li. Simvp: Simpler yet better video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3170–3180, 2022. 2
- [14] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376, 2006. 2
- [15] Aiming Hao, Yuecong Min, and Xilin Chen. Self-mutual distillation learning for continuous sign language recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11303–11312, 2021. 1, 2, 3, 4, 5, 6
- [16] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9729–9738, 2020. 5
- [17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 4
- [18] Yi Ji, Chunping Liu, Shengrong Gong, and Weidong Cheng. 3d hand gesture coding for sign language learning. In *2016 International Conference on Virtual Reality and Visualization (ICVRV)*, pages 407–410. IEEE, 2016. 2
- [19] Suhail Muhammad Kamal, Yidong Chen, Shaozi Li, Xiaodong Shi, and Jiangbin Zheng. Technical approaches to chinese sign language processing: a review. *IEEE Access*, 7:96926–96935, 2019. 1, 2, 8
- [20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6
- [21] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3
- [22] Oscar Koller, Necati Cihan Camgoz, Hermann Ney, and Richard Bowden. Weakly supervised learning with multi-stream cnn-lstm-hmms to discover sequential parallelism in sign language videos. *IEEE transactions on pattern analysis and machine intelligence*, 42(9):2306–2320, 2019. 1, 3, 6
- [23] Oscar Koller, Jens Forster, and Hermann Ney. Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. *Computer Vision and Image Understanding*, 141:108–125, 2015. 1, 5
- [24] Oscar Koller, Sepehr Zargaran, and Hermann Ney. Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4297–4305, 2017. 1, 3, 5
- [25] Siyuan Li, Zedong Wang, Zicheng Liu, Cheng Tan, Haitao Lin, Di Wu, Zhiyuan Chen, Jiangbin Zheng, and Stan Z Li. Efficient multi-order gated aggregation network. *arXiv preprint arXiv:2211.03295*, 2022. 1
- [26] Zhi-jie Liang, Sheng-bin Liao, and Bing-zhang Hu. 3d convolutional neural networks for dynamic sign language recognition. *The Computer Journal*, 61(11):1724–1736, 2018. 2
- [27] Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. 2018. 6

- [28] Yuecong Min, Aiming Hao, Xiujuan Chai, and Xilin Chen. Visual alignment constraint for continuous sign language recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11542–11551, 2021. [2](#), [3](#), [4](#), [5](#)
- [29] Zhe Niu and Brian Mak. Stochastic fine-grained labeling of multi-state sign glosses for continuous sign language recognition. In *European Conference on Computer Vision*, pages 172–186. Springer, 2020. [1](#), [2](#), [3](#), [5](#), [6](#)
- [30] Tse-Yu Pan, Li-Yun Lo, Chung-Wei Yeh, Jhe-Wei Li, Hou-Tim Liu, and Min-Chun Hu. Sign language recognition in complex background scene based on adaptive skin colour modelling and support vector machine. *International Journal of Big Data Intelligence*, 5(1-2):21–30, 2018. [2](#)
- [31] Jinye Peng, Li Yulong, et al. Chinese sign language recognition based on gray-level co-occurrence matrix and other multi-features fusion. In *2009 4th IEEE Conference on Industrial Electronics and Applications*, pages 1569–1572. IEEE, 2009. [2](#)
- [32] Junfu Pu, Wengang Zhou, Hezhen Hu, and Houqiang Li. Boosting continuous sign language recognition via cross modality augmentation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1497–1505, 2020. [1](#), [5](#)
- [33] Junfu Pu, Wengang Zhou, and Houqiang Li. Iterative alignment network for continuous sign language recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4165–4174, 2019. [1](#), [2](#), [5](#)
- [34] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. [1](#), [2](#), [5](#)
- [35] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pages 1278–1286. PMLR, 2014. [3](#)
- [36] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. [8](#)
- [37] Cheng Tan, Zhangyang Gao, Siyuan Li, Yongjie Xu, and Stan Z Li. Temporal attention unit: Towards efficient spatiotemporal predictive learning. *arXiv preprint arXiv:2206.12126*, 2022. [2](#)
- [38] Cheng Tan, Jun Xia, Lirong Wu, and Stan Z Li. Co-learning: Learning from noisy labels with self-supervision. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1405–1413, 2021. [2](#)
- [39] Yiqi Tong, Jiangbin Zheng, Hongkang Zhu, Yidong Chen, and Xiaodong Shi. A document-level neural machine translation model with dynamic caching guided by theme-rheme information. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4385–4395, 2020. [1](#)
- [40] Hanjie Wang, Xiujuan Chai, Xiaopeng Hong, Guoying Zhao, and Xilin Chen. Isolated sign language recognition with grassmann covariance matrices. *ACM Transactions on Accessible Computing (TACCESS)*, 8(4):1–21, 2016. [2](#)
- [41] Hanjie Wang, Xiujuan Chai, Yu Zhou, and Xilin Chen. Fast sign language recognition benefited from low rank approximation. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 1, pages 1–6. IEEE, 2015. [2](#)
- [42] Chong Wu, Jiangbin Zheng, Zhenan Feng, Houwang Zhang, Le Zhang, Jiawang Cao, and Hong Yan. Fuzzy slic: Fuzzy simple linear iterative clustering. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(6):2114–2124, 2020. [1](#)
- [43] Jiangbin Zheng, Yidong Chen, Chong Wu, Xiaodong Shi, and Suhail Muhammad Kamal. Enhancing neural sign language translation by highlighting the facial expression information. *Neurocomputing*, 464:462–472, 2021. [1](#), [3](#)
- [44] Jiangbin Zheng, Siyuan Li, Cheng Tan, Chong Wu, Yidong Chen, and Stan Z Li. Leveraging graph-based cross-modal information fusion for neural sign language translation. *arXiv preprint arXiv:2211.00526*, 2022. [3](#)
- [45] Jiangbin Zheng, Yile Wang, Ge Wang, Jun Xia, Yufei Huang, Guojiang Zhao, Yue Zhang, and Stan Z Li. Using context-to-vector with graph retrofitting to improve word embeddings. *arXiv preprint arXiv:2210.16848*, 2022. [5](#)
- [46] Jiangbin Zheng, Zheng Zhao, Min Chen, Jing Chen, Chong Wu, Yidong Chen, Xiaodong Shi, and Yiqi Tong. An improved sign language translation model with explainable adaptations for processing long sign sentences. *Computational Intelligence and Neuroscience*, 2020, 2020. [3](#)
- [47] Hao Zhou, Wengang Zhou, and Houqiang Li. Dynamic pseudo label decoding for continuous sign language recognition. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1282–1287. IEEE, 2019. [5](#)
- [48] Hao Zhou, Wengang Zhou, Yun Zhou, and Houqiang Li. Spatial-temporal multi-cue network for continuous sign language recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13009–13016, 2020. [1](#), [3](#), [5](#), [6](#)
- [49] Yongjie Zhuang, Bo Lv, Xinjun Sheng, and Xiangyang Zhu. Towards chinese sign language recognition using surface electromyography and accelerometers. In *2017 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP)*, pages 1–5. IEEE, 2017. [2](#)
- [50] Ronglai Zuo and Brian Mak. C2slr: Consistency-enhanced continuous sign language recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5131–5140, 2022. [1](#), [2](#), [3](#), [5](#), [6](#)