Improving Sign Language Translation with Monolingual Data by Sign Back-Translation

Hao Zhou[1] Wengang Zhou[1,2,*] Weizhen Qi[1] Junfu Pu[1] Houqiang Li[1,2,*]
[1] CAS Key Laboratory of GIPAS, EEIS Department, University of Science and Technology of China
[2] Institute of Artificial Intelligence, Hefei Comprehensive National Science Center
zhouh156@mail.ustc.edu.cn, zhwg@ustc.edu.cn, {weizhen,pjh}@mail.ustc.edu.cn, lihq@ustc.edu.cn

Abstract

Despite existing pioneering works on sign language translation (SLT), there is a non-trivial obstacle, i.e., the limited quantity of parallel sign-text data. To tackle this parallel data bottleneck, we propose a sign back-translation (SignBT) approach, which incorporates massive spoken language texts into SLT training. With a text-to-gloss translation model, we first back-translate the monolingual text to its gloss sequence. Then, the paired sign sequence is generated by splicing pieces from an estimated gloss-to-sign bank at the feature level. Finally, the synthetic parallel data serves as a strong supplement for the end-to-end training of the encoder-decoder SLT framework.

To promote the SLT research, we further contribute CSL-Daily, a large-scale continuous SLT dataset. It provides both spoken language translations and gloss-level annotations. The topic revolves around people's daily lives (e.g., travel, shopping, medical care), the most likely SLT application scenario. Extensive experimental results and analysis of SLT methods are reported on CSL-Daily. With the proposed sign back-translation method, we obtain a substantial improvement over previous state-of-the-art SLT methods.

1. Introduction

Sign language serves as the primary communication method among the deaf community. However, in a society where the spoken language is primarily used, the deaf people face issues of social isolation and communication barrier in daily lives [4]. Due to the significant social impact and the cross-modality challenge, sign language understanding has been attracting more and more research attention [1, 4, 10, 17, 22, 26, 37, 48]. In this paper, we concentrate on sign language translation (SLT), which aims to automatically generate the spoken language translation from a continuous sign video.

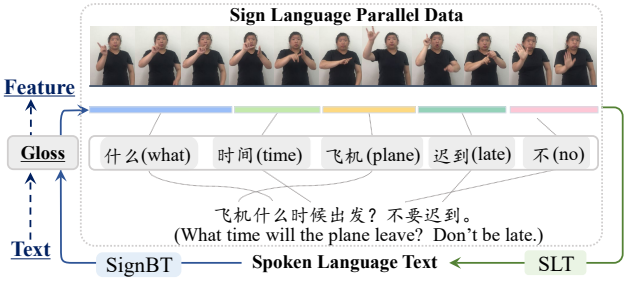

Figure 1. Pipelines of sign language translation (SLT) and sign back-translation (SignBT). Our SignBT approach establishes an inverse path of SLT and uses it to enrich text-feature pairs from external monolingual data for SLT training.

Considering the different grammar rules and vocabularies of sign language and spoken language, SLT is typically treated as a sequence-to-sequence learning problem. Existing SLT systems typically rely on the encoder-decoder architectures [10, 11, 25]. Despite the success of encoder-decoder networks in neural machine translation (NMT), the translation quality in SLT is limited, which is partially attributed to the huge gap in the training data size. While the News Translation task [3] provides over 77M English-German data, the only suitable SLT dataset PHOENIX-2014T [10] has less than 9K Sign-German data. To alleviate it, there are two possible solutions, *i.e.*, collecting millions of parallel pairs or introducing monolingual data. The high charge of sign video collection and annotation makes the former a luxury. In contrast, making good use of accessible monolingual texts is a promising direction for SLT.

In this work, we propose to generate synthetic parallel data with monolingual texts for SLT training. Our method is inspired by the success of text-to-text back-translation in NMT [36]. They train an inverse model with available pairs and use it to back-translate the monolingual data. However, when it goes to SLT, the key challenge becomes how to bridge the huge domain gap between text and vision signals. A straightforward idea is to generate the sign video from a

*Corresponding authors: Wengang Zhou and Houqiang Li
http://home.ustc.edu.cn/~zhouh156/dataset/csl-daily

sentence, which, however is a more challenging task involving various immature techniques, such as skeleton prediction [35], gesture generation [12] and temporal coherence fidelity [38]. A compromise option is to regress the feature sequence of video frames from a sentence. Unfortunately, it is an indeterminate problem and hard to formulate, because one sentence may correspond to numerous possible feature sequences, since the feature space of sign videos is far larger than the combination space of text vocabulary.

To avoid the above problems, we propose a two-stage sign back-translation (SignBT) approach: text-to-gloss and gloss-to-sign. It is formulated as an inverse problem of SLT with an additional signal "gloss" (see Figure 1). Gloss is a token of sign language word, which is annotated along with the order of signs in a video with no clear boundaries. We first train a text-to-gloss translator with available text-gloss pairs and predict the gloss sequence for each monolingual text. Then, to achieve the sequence-level gloss-to-sign conversion, we adopt a primitive but effective method, splicing sign pieces from features of segmented videos, which is somewhat analogous to concatenative text-to-speech synthesis [18, 43]. To acquire the precise boundary of each gloss, we train a sign-to-gloss network with connectionist temporal classification (CTC) [15] and find the most likely alignment path for segmentation. The sign pieces could be segmented and stored as a sign bank in advance. Finally, we simplify the whole process into a text-to-text back-translation problem and a sequence splicing operation from pieces in the bank.

The key reason that the synthetic data benefits SLT training lies in the two aspects of realism, *i.e*., the target text from the real language corpus and the source sign sequence spliced from the real feature bank. Though the fake pair may not be perfect as a real training data, it helps regularize the decoder when speaking target language and improve the robustness of extracting information from the source. Through extensive experiments, we verify the significant improvement of SLT models brought by monolingual data.

Acquiring high-quality corpus is always crucial for SLT. In this paper, we provide the first large-scale Chinese Sign Language Translation benchmark, CSL-Daily. The native expression, compact annotation and clear hand details make our corpus suitable for a series of sign language research, *e.g*., sign language recognition, translation and generation. The evaluations on CSL-Daily of various SLT baselines are reported with in-depth analysis.

Our main contributions are summarized as follows,

- We propose a sign back-translation approach to tackle parallel data shortage in SLT.
- We contribute a new large-scale SLT benchmark with rich contents and compact annotations.
- Extensive experiments on two datasets demonstrate the effectiveness of our SignBT mechanism.

2. Related Work

Sign Language Recognition. Sign language recognition (SLR) includes two sub-tasks: isolated SLR and continuous SLR. While isolated SLR aims to recognize one sign from a trimmed video, continuous SLR tries to recognize an ordered sign gloss sequence from a continuous video. Early works in isolated SLR utilize hand-crafted features [31, 41] for sign description. With the success of deep learning, 2D and 3D convolutional neural networks (CNN) [6, 19, 39] achieve favorable performance on action related tasks [6]. It inspires more research groups to study continuous SLR with large-scale vocabulary [13, 22, 34]. To enable end-to-end training, connectionist temporal classification (CTC) [15] is widely adopted by continuous SLR models [7, 14, 30, 33, 49]. With the development of neural machine translation, Camgöz *et al*. formulate a new task, neural sign language translation (SLT) [10], which is becoming an active and promising direction [11, 25].

Sign Language Translation. SLT is different from SLR mainly in the aspect of sequence learning. The encoder-decoder based methods [2, 28, 42] are adopted to process the difference in word order and vocabulary between sign language and spoken language. In [10], Camgöz *et al*. propose an SLT dataset PHOENIX-2014T and provide spoken language annotations. They use attention-based encoder-decoder models to learn how to translate from spatial representations or sign glosses. Recently, transformer networks [44] have been popular for neural machine translation (NMT). Camgöz *et al.* [11] apply transformers into sequence learning of sign language. Their work explores the multi-task formulation of continuous SLR and SLT. Under transformer framework, Li *et al*. [25] explore the hierarchical structure in sign video representation. Besides, some works [5, 54] improve the SLT framework by considering the multi-cue characteristic of sign language.

Monolingual Data Exploration. The integration of monolingual data for neural machine translation (NMT) models is first investigated in [16]. Gulcehre *et al*. train the NMT model independently and use a language model from monolingual data for re-scoring during the decoding process. To introduce monolingual data in model training, Sennrich *et al*. propose a back-translation approach [36] to generate synthetic parallel data for training without changing the encoder-decoder structure. In [8], sentences with blank facetracks are utilized to enhance the decoder of lip reading. Different from previous work, we design a sign back-translation mechanism across domains of video and language, which brings state-of-the-art results in SLT datasets and new insight to approach SLT.

Sign Language Dataset. High-quality datasets are essential in promoting sign language research. A summary of publicly available datasets for video-based sign language research is presented in Table 1. The majority of them are

Table 1. Summary of public available video-based sign language benchmarks popular for computer vision research. (SignDict: the corpus has isolated or segmented sign videos as a dictionary. Continuous: the corpus is composed of videos of continuous sign sentences and gloss-level annotations. Translation: the corpus has spoken language translation annotations.)

Dataset	Language	Attribute				Statitics			Source
		SignDict	Continuous	Translation	Resolution	#Signs	#Videos (avg. signs)	#Signers	
DEVISIGN [48]	CSL	✓			-	2,000	24,000 (1)	8	Lab
ICSL [52]	CSL	✓			1280×720	500	125,000 (1)	50	Lab
MSASL [20]	ASL	✓			-	1,000	25,513 (1)	222	Web
WLASL [24]	ASL	✓			-	2,000	21,083 (1)	119	Web
BSL-1K [1]	BSL	✓			-	1,064	273,000 (1)	40	TV
INCLUDE [40]	ISL	✓			1920×1080	263	4,287 (1)	7	Lab
PHOENIX-2014 [23]	DGS		✓		210×260	1,081	6,841 (11)	9	TV
CCSL [17]	CSL	✓	✓		1280×720	178	25,000 (4)	50	Lab
SIGNUM [47]	DGS	✓	✓	✓ (German)	776×578	455	15,075 (7)	25	Lab
PHOENIX-2014T [10]	DGS		✓	✓ (German)	210×260	1,066	8,257 (9)	9	TV
CSL-Daily (ours)	CSL	✓	✓	✓ (Chinese)	1920×1080	2,000	20,654 (7)	10	Lab

composed of word-level sign videos. To achieve continuous SLR evaluation, some datasets provide gloss-level annotations [17, 23, 47]. Although German translations are provided in SIGNUM, it is not appropriate for SLT tasks, due to its limited vocabulary and sentences. Hence, PHOENIX-2014T [10] becomes the only suitable dataset for SLT research [11, 25]. However, the frames in [23] are cropped from a specific TV program of the weather forecast and thus in a low resolution. It constrains the exploration of language model and sign language generation in hand details. As a considerable complement, our CSL-Daily contains over 20K 1080P sign videos. It provides both gloss and translation annotations and covers diverse topics of daily lives.

3. Proposed Method

Given a sign video $\mathbf{x} = \{x_t\}_{x=1}^{T}$ with T Frames, sign language translation (SLT) can be formulated as learning the conditional probability $p(\mathbf{y}|\mathbf{x})$ of generating a spoken language sentence $\mathbf{y} = \{y_u\}_{u=1}^{U}$ with U words. In addition, existing SLT datasets provide gloss-level annotations for pre-training of sign embedding networks. Unlike spoken language which is non-monotonic to sign language, the gloss-level annotation $\mathbf{g} = \{g_v\}_{v=1}^{V}$ with V sign glosses is order-consistent with sign gestures. An overview of our framework for SLT is depicted in Figure 2.

The remainder of this section is organized as follows. In subsection 3.1, we elaborate the building method of our sign bank with a pre-trained sign embedding network. Then, the transformer-based SLT framework is revisited. Finally, we detail our sign back-translation process in subsection 3.2.

3.1. Sign Bank Generation

To acquire the gloss-to-sign mapping, we are dedicated to build a sign bank containing video piece features indexed by its gloss vocabulary. However, due to the high cost of hiring experts, existing continuous sign datasets only have sentence-level gloss annotations [10, 23, 52] without the boundary ground-truth. It hinders the segmentation of sign feature sequences for our sign bank. Hence, we propose to

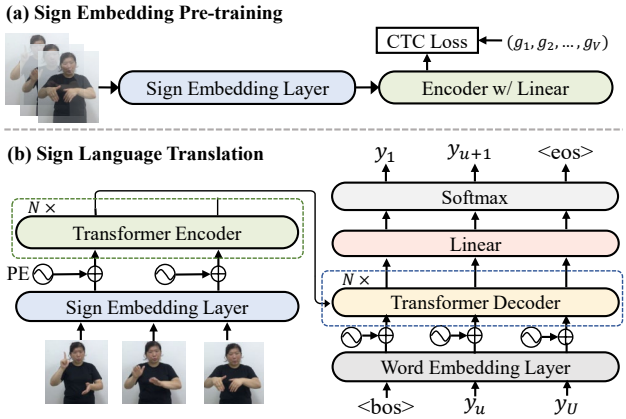

Figure 2. An overview of our SLT framework. We show the sign embedding pre-training process in (a). It is trained with CTC Loss and gloss-level annotations. The detailed encoder-decoder structure for SLT is shown in (b). (PE: Positional Encoding.)

establish the sign bank with estimated alignment paths from a pre-trained sign embedding network.

Sign Embedding Layer. Unlike word embedding technique in NMT, which serves for word association learning, sign embedding is to convert a series of video frames to its feature representation. Our sign embedding layer Ω adopts a combination of 2D and 1D CNNs for spatiotemporal encoding [14]. In this work, the embedding operation is performed in the clip-level. We split video frames $\mathbf{x}$ into N clips $\mathbf{c} = \{c_n\}_{n=1}^{N}$. The number of clips is $N = \lceil \frac{T}{s} \rceil$ with sliding window size w and stride size s. By passing clips through Ω, the embeddings $\mathbf{f} = \{f_n\}_{n=1}^{N}$ are extracted as follows,

$$f_n = \text{SignEmbedding}(c_n) = \Omega_\theta(c_n), \tag{1}$$

where θ denotes the parameters of the CNN network.

Sign-to-Gloss Pre-Training. The embedding layer is usually pre-trained with gloss-level annotations [10, 11]. For our embedding layer, we use connectionist temporal classification [15] (CTC) with a transformer encoder [45] for gloss-level temporal modelling. The gloss probabilities

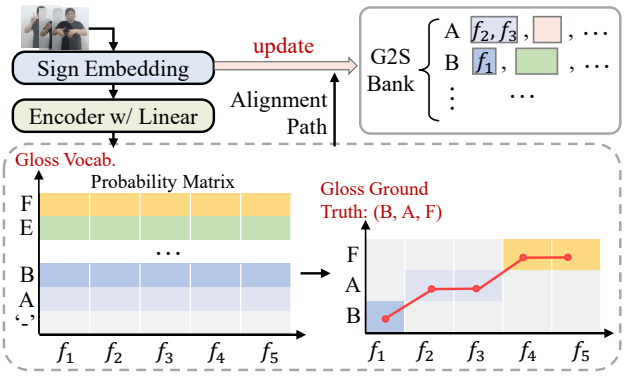

Figure 3. Illustration of constructing a Gloss-to-Sign (G2S) Bank according to the most probable alignment paths.

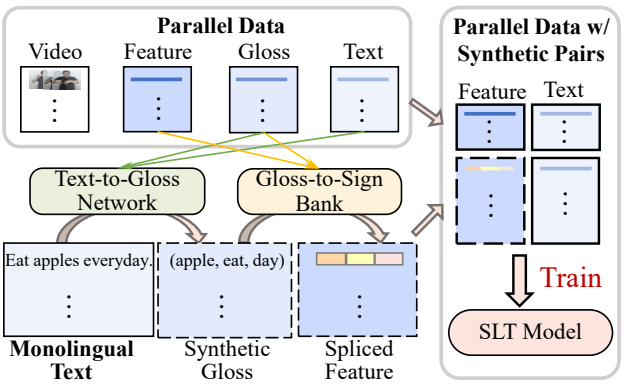

Figure 4. Illustration of the sign back-translation process.

$p(g_n|\mathbf{f})$ at the n-th time step could be estimated by a linear layer with softmax activation. According to CTC, the conditional probability $p(\mathbf{g}|\mathbf{x})$ is computed as the sum of probabilities of all feasible paths,

$$p(\mathbf{g}|\mathbf{x}) = \sum_{\pi \in \mathcal{B}^{-1}(\mathbf{g})} p(\pi|\mathbf{f}), \tag{2}$$

where π is a sign-to-gloss alignment path and $\mathcal{B}$ denotes the mapping between them. The embedding layer is trained through the CTC Loss $L_{\text{ctc}} = -\ln p(\mathbf{g}|\mathbf{x})$.

Gloss-to-Sign Bank. Given a sign embedding sequence $\mathbf{f} = \{f_n\}_{n=1}^{N}$ extracted from Ω and its corresponding gloss sequence $\mathbf{g} = \{g_v\}_{v=1}^{V}$, we find the most probable alignment path $\hat{\pi}$ between them as follows,

$$\hat{\pi} = \underset{\pi \in \hat{\mathcal{B}}^{-1}(\mathbf{g})}{\arg\max}\, p(\pi|\mathbf{f}). \tag{3}$$

The searching space is constrained within paths that conform to the mapping function $\hat{\mathcal{B}}$ with no blank labels (See Figure 3). Notably, paths going through the blank label [15] are excluded from the searching space to ensure the proper length as a sign sentence after splicing. The searching problem could be accelerated with Viterbi algorithm[46, 53].

With the estimated alignments, we segment the embedding sequence of each video into gloss pieces. They constitute a gloss-to-sign (G2S) bank in embedding space with a look-up table which is indexed by the gloss vocabulary. Each gloss slot may have multiple feature pieces.

3.2. SLT Training with Monolingual Data

Comparing with the limited size of sign-text pairs, the monolingual spoken language corpus is easy to reach the volume of millions. To make use of the monolingual texts, we propose to establish an inverse path of SLT and use it to enrich parallel data for training.

Sign Language Translation. The encoder-decoder framework is widely utilized and explored in SLT [10, 11]. Here, we briefly introduce the transformer-based encoder-decoder structure in our SLT framework (see Figure 2b). *Notably*, our approach is not limited to this architecture.

The encoder is composed of several stacked identical layers. Each layer has a self-attention network and a feed-forward network. To provide sequential clues, the input of the first layer is summed with a positional encoding (PE) vector as $\hat{f}_n = f_n + \text{PE}(n)$. The encoder takes all encoded input $\hat{\mathbf{f}}$ and generates N hidden vectors as follows,

$$h_{1:N} = \text{Encoder}(\hat{f}_{1:N}). \tag{4}$$

During decoding, we first pass each word y_u through a lookup table for word embedding as follows,

$$w_u = \text{WordEmbedding}(y_u). \tag{5}$$

Here, $\hat{w}_u = w_u + \text{PE}(u)$ is the positionally encoded word embedding of y_u. The decoder network includes an extra layer which performs attention operation over the encoder hidden vectors $h_{1:N}$ and hidden states of previously predicted words, for information aggregating. Then, the probability of words at the u-th step is generated as follows,

$$o_u = \text{Decoder}(\hat{w}_{1:u-1}, h_{1:N}), \tag{6}$$
$$z_u = \text{softmax}(W o_u + b). \tag{7}$$

The initial word is $<bos>$ which indicates the beginning of a sentence. Finally, we compute the conditional probability of $p(\mathbf{y}|\mathbf{x})$ as follows,

$$p(\mathbf{y}|\mathbf{x}) = \prod_{u=1}^{U} p(y_u|y_{1:u-1}, \mathbf{x}) = \prod_{u=1}^{U} z_{u,y_u}. \tag{8}$$

To optimize the whole structure, the objective function is formulated as $L_{\text{SLT}} = -\ln p(\mathbf{y}|\mathbf{x})$. During inference, the words in spoken language text are predicted word-by-word as in Equation 6. The beam search [50] strategy is used to estimate a better decoding path in an acceptable range.

Sign Back-Translation. Given an SLT corpus, parallel pairs of sign videos $\mathcal{X}$ and spoken language texts $\mathcal{Y}$ are converted to $(\mathcal{F}, \mathcal{Y})$ pairs through a sign embedding network.

Table 2. Key statistics of the CSL-Daily split. (OOV: out-of-vocabulary, *e.g.*, words that occur in Dev set but not in Train set. Singleton: words that only occur once in Train unique sentences.)

	Sign Gloss			Chinese		
	Train	Dev	Test	Train	Dev	Test
segments	18,401	1,077	1,176	⟵ same		
duration (h)	20.62	1.24	1.41	⟵ same		
frames	2,227,178	134,530	153,074	⟵ same		
vocab. size	2,000	1,344	1,345	2,343	1,358	1,358
total words/chars	133,714	8,173	9,002	291,048	17,304	19,288
total OOVs	-	0	0	-	64	69
unique sentences	6,598	797	798	6,598	797	798
singletons	247	-	-	418	-	-

Meanwhile, monolingual spoken language texts $\mathcal{Y}'$ are collected, sharing a similar vocabulary with $\mathcal{Y}$. The following target is to generate synthetic pairs $(\mathcal{F}'_{\text{syn}}, \mathcal{Y}')$ with monolingual data $\mathcal{Y}'$, as depicted in Figure 4.

First, we train a text-to-gloss (T2G) network with existing parallel pairs of $(\mathcal{Y}, \mathcal{G})$ for back-translation. Then, the collected spoken language texts $\mathcal{Y}'$ are first translated to sign gloss texts $\mathcal{G}'_{\text{syn}}$. We splice gloss pieces from the G2S bank into sign embedding sequences $\mathcal{F}'_{\text{syn}}$ according to $\mathcal{G}'_{\text{syn}}$. As each gloss may have multiple feature pieces in G2S bank, we randomly sample one piece from them for splicing. In different training epochs, the spliced feature sequences of the same synthetic gloss sequence are different due to the random selections. It largely enriches the feature combinations in the source domain.

Finally, we mix synthetic pairs $(\mathcal{F}'_{\text{syn}}, \mathcal{Y}')$ with annotated pairs $(\mathcal{F}, \mathcal{Y})$ together for SLT training. Notably, the texts in the decoder side always comes from a real corpus.

4. The Proposed CSL-Daily Dataset

CSL-Daily aims to offer the community a new large-scale sign language corpus, which is appropriate for both practical application and academic research. In this section, the details of dataset production are elaborated.

4.1. Data Collection

The content of our corpus mainly revolves around the daily life of the deaf community. It covers a wide range of topics, including family life, medical care, school life, bank service, shopping, social contact and so on.

Deaf community involvement is essential in developing sign language corpus [4]. We invite a professional team composed of an expert in the field of sign language linguistics and several sign language teachers to help design the specific content and produce reference texts for recording guidance. The texts are mainly collected from some Chinese Sign Language textbooks and test material, and partly from some Chinese corpora.

There are 10 signers participating in the video recording work. They are all native signers from the deaf community and 4 of them are engaged in sign language education. To remove the ambiguity of meanings, sign videos of one senior signer are recorded in advance as reference. After watching guidance videos, each reference text is signed again by one or two signers. No signers sign the same reference text twice. The requirement for signers is to ensure the natural expression of sign language and describe the content in reference texts as fully as possible.

Table 3. Statistics of the training data. (OOV-%: the ratio of words or characters which are out of the parallel data vocabulary.)

	Amount	OOV (%)	Source
DGS↔ German	7,096	-	PHOENIX-2014T
German texts	212,247	7.07%	Wiki, weather forecast website
CSL↔Chinese	18,402	-	CSL-Daily
Chinese texts	566,682	1.80%	Wiki, WebText in CLUE [51]

The resolution of recorded videos is 1920×1080 and the frame rate is 30 FPS. The motionless frames in the beginning and the end of a video are cut off carefully.

4.2. Annotation

Our CSL-Daily provides two levels of annotation, *i.e.*, sign gloss and spoken language translation. Our annotation work relies on the cooperation of senior native signers and authors of this work. First, each sign running in videos is annotated with a Chinese word which has the similar meaning. Then, we adopt two strategies to merge the sign gloss with the same visual expression. One is to check the glosses with the similar meaning. The other is to train and test a sign-to-gloss network on the dataset. With the confusion matrix of predicted gloss, we focus the top-k confused pairs and check if they share the same sign indeed. With three rounds of double-checking, we reduce the vocabulary size of annotated sign glosses from $>$ 3k to 2k. Then, spoken language translation annotation is conducted according to original reference texts and sign gloss annotations.

The detailed statistics of the dataset is shown in Table 2. In addition, a sign dictionary (SignDict) is produced. Each non-single sign is recorded by 4 sign teachers. The SignDict can be used for tasks like sign spotting, sign segmentation, isolated SLR and gloss-free SLT in the future. It can also serve as a reference collection for qualitative analysis of continuous sign language related tasks.

5. Experiments

5.1. Experimental Setup

Dataset. We mainly conduct ablation studies and evaluate our method on CSL-Daily. Experimental analysis on PHOENIX-2014T [10] is also reported. PHOENIX-2014T is a large-scale SLT corpus composed of German Sign Language (Deutsche Gebärdensprache, DGS). It is an extended version of PHOENIX-2014 [23] and contains parallel sign videos, gloss annotations and their German translations. The split of videos for Train, Dev and Test is 7096, 519

Table 4. Temporal Inception Network Architecture [14] (TIN) for sign language embedding. 1D Batch Norm (BN) layer is added after each temporal convolution layer.

Layer	Stride	Kernel	Output Size
Input	-	-	$T \times 224 \times 224 \times 3$
Inception Blocks w/ BN	$1, 32, 32$	-	$T \times 7 \times 7 \times 1024$
Global AvgPooling2D	$1, 7, 7$	$1, 7, 7$	$T \times 1024$
Conv1D-BN1D-ReLU	$1, 1, 1$	$5, 1, 1$	$T \times 512$
MaxPooling1D	$2, 1, 1$	$2, 1, 1$	$(T/2) \times 512$
Conv1D-BN1D-ReLU	$1, 1, 1$	$5, 1, 1$	$(T/2) \times 512$
MaxPooling1D	$2, 1, 1$	$2, 1, 1$	$(T/4) \times 512$
Transformer Encoder	-	-	$(T/4) \times 512$
Fully Connection	-	-	$(T/4) \times C$

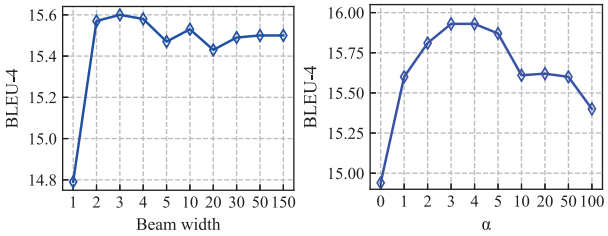

Figure 5. The effect of beam width and length penalty α on CSL-Daily Dev set under S2G2T setting.

and 642, respectively. The vocabulary size is 1115 for sign gloss and 3000 for German.

Training Data. Except the sentences in datasets, the majority are from the open wikipedia corpus [3] (see Table 3). To be close to the datasets' topic, we also collect some texts from a German weather forecast website and extract a subset about trivia in daily lives from CLUE corpus [51].

Evaluation. To assess the sign embedding layer, we adopt Word Error Rate (WER) as the metric of measuring the similarity between the predicted gloss sequence and the ground truth. To measure the SLT performance, we select BLEU [32] and ROUGE [27] scores, commonly used in NMT. Here, BLEU is calculated with n-grams from 1 to 4. ROUGE refers to ROUGE-L F1-Score [27].

Sub-Problem Definition. In this paper, we mainly discuss two sub-problems of SLT as follows,

1. Sign-to-Text (S2T): It predicts the spoken language translations directly from sign embedding sequences in an end-to-end pipeline.
2. Sign-to-Gloss-to-Text (S2G2T): It resorts to sign gloss as an intermediate state. The G2T network is trained with sign glosses predicted from an S2G network.

5.2. Implementation Details

Sign Embedding Layer. The input frames are resized to 224×224. For data augmentation, we use random shift and random discard or copy of 20% frames. The architecture of our sign embedding layer is presented in Table 4. The pre-trained weights on ImageNet are loaded for initialization. The encoder with a classifier is only for pre-training and will be discarded in the following SLT experiments.

Table 5. Evaluation of the S2G network combinations on WER (the lower the better).

S2G combination		PH2014T		CSL-Daily	
Sign Embedding	Encoder	Dev	Test	Dev	Test
I3D	Transformer	32.6	33.2	45.4	44.3
TIN	Transformer	26.2	27.5	36.1	35.7
BN-TIN	Transformer	23.0	24.1	33.6	33.1
BN-TIN	Conv1D	24.7	25.1	33.4	33.3
BN-TIN	Bi-GRU	**22.7**	**23.9**	**33.2**	**32.2**

Table 6. Performance of CSLR baselines on CSL-Daily. $*$ denotes that the results are based on our implementation.

Method	Dev		Test	
	del/ins	WER	del/ins	WER
SubUNets [9]	14.8/3.0	41.4	14.6/2.8	41.0
LS-HAN* [17]	14.6/5.7	39.0	14.8/5.0	39.4
TIN-Iterative* [14]	12.8/3.3	**32.8**	12.5/2.7	32.4
Joint-SLRT [11]	10.3/4.4	33.1	9.6/4.1	**32.0**
FCN-GFE* [7]	12.8/4.0	33.2	12.6/3.7	32.5
BN-TIN+Transf. Encoder	13.9/3.4	33.6	13.5/3.0	33.1

Transformer. In our experiments, the setting of all transformer layers is the same. The hidden size is 512 and the feed-forward size is 2048. Each layer has 8 attention heads which is the basic setting of transformer [44]. The dropout rate is all set to 0.1 to alleviate over-fitting.

Optimization. The sign embedding layer is trained end-to-end under CTC Loss with batch size 2. No iterative training [53], online refining [7] or temporal sampling [30] are used. We use Adam optimizer [21] and set the weight decay to 1×10^{-6}. The learning rate is initialized as 5×10^{-5}. It will be reduced by a factor of 0.5 until 2×10^{-6} when WER of Dev stops decreasing for 3 epochs. Experiments are run on 4 Titan RTX GPUs.

The transformer is trained end-to-end under masked cross-entropy loss [45] with batch size 32. The rate of label smoothing [29, 45] is 0.1. We use Adam optimizer with no weight decay. The learning rate is fixed to 5×10^{-5}. Experiments are run on 1 Titan RTX GPU.

Inference. For decoding in the inference process, we use the beam search strategy [50]. It is combined with a length penalty α [50] for length normalization. For PHOENIX-2014T, we set beam width to 3 and α to 1, following [11]. In contrast, Chinese sentences are longer in character-level tokenization. We search the combinations in Figure 5 and use beam width of 3 and length penalty α of 3.

5.3. Ablation Study

The ablation experiments are mainly conducted on CSL-Daily-Dev, presenting the characteristics of this new corpus.

Sign Language Embedding. In Table 5, we investigate which kind of spatiotemporal combinations is suitable for sign embedding. The I3D model [6] achieves good performance in the action recognition task. However, with less spatial details, it still has a performance gap to 2D-CNN based methods. Unlike previous re-fining methods [7, 14, 53], we use 1D batch normalization (BN) to

Table 7. Evaluation of different encoder-decoder frameworks on CSL-Daily. (R: ROUGE, B-n: BLEU-n, the higher the better.)

S2G2T	R	B-1	B-2	B-3	B-4
seq2seq w/ Bahdanau [2]	39.63	41.58	25.34	16.08	10.63
seq2seq w/ Luong [28]	40.18	41.46	25.71	16.57	11.06
Transformer [45]	**44.21**	**46.61**	**32.11**	**22.44**	**15.93**
S2T	R	B-1	B-2	B-3	B-4
seq2seq w/ Bahdanau [2]	33.83	33.99	19.48	11.66	7.11
seq2seq w/ Luong [28]	34.28	34.22	19.72	12.24	7.96
Transformer [45]	**37.29**	**40.66**	**26.56**	**18.06**	**12.73**

Table 8. The number of epochs for warm-up on CSL-Daily. (Warm-up: mix all synthetic data with parallel data for training)

warm-up	S2G2T				S2T			
#epochs	R	B-2	B-3	B-4	R	B-2	B-3	B-4
0 (0.4h)	44.21	32.11	22.44	15.93	37.29	26.56	18.06	12.73
1 (0.6h)	46.22	34.47	24.70	18.06	42.69	31.72	22.03	15.64
5 (1.6h)	47.68	35.51	25.58	18.73	46.56	34.33	24.54	17.98
10 (2.9h)	47.88	36.08	26.20	19.38	47.75	35.17	25.58	19.11
20 (5.4h)	48.01	35.57	25.82	19.18	48.55	36.07	26.24	19.61
50 (12.9h)	**48.38**	**36.16**	**26.26**	**19.53**	48.77	36.63	26.90	20.20
100 (25.4h)	47.83	36.05	26.17	19.42	**49.09**	**36.91**	**27.20**	**20.50**

Table 9. The ratio of synthetic data to parallel data for the training process after warm-up.

	S2G2T				S2T			
ratio	R	B-2	B-3	B-4	R	B-2	B-3	B-4
0.0: 1	**48.38**	**36.16**	**26.26**	**19.53**	48.77	36.63	26.90	20.20
0.1: 1	46.97	35.49	25.80	19.20	**49.49**	**37.23**	**27.51**	**20.80**
0.5: 1	46.30	34.77	25.19	18.82	49.15	36.88	27.23	20.64
1.0: 1	45.76	34.43	24.65	18.22	49.21	36.38	26.80	20.27

mitigate the unstable activation in temporal structures. It achieves favorable performance under end-to-end training without bells and whistles. Hence, we adopt BN-TIN as our sign embedding layer. In Table 6, we also provides some results of CSLR methods on CSL-Daily for reference.

Encoder-Decoder Framework. In Table 7, we evaluate encoder-decoder networks with different architectures. For the sophisticated design of self-attention [45], the transformer-based SLT model achieves obvious advantage over previous recurrent neural network-based methods [2, 28]. We set the transformer-based network as our baseline model for the following experiments.

The Participation of Synthetic Data. We generate synthetic pairs from texts in Table 3. They are over 30 times the amount of annotated parallel data. If directly mixing them for training with no adjustment, the noise in synthetic pairs will largely disturb the model learning. Hence, we first use all data for warm-up and then train models until convergence with less synthetic pairs.

In Table 8, we evaluate the SLT performance with different warm-up epochs on CSL-Daily. Even with only one warm-up epoch, the performance gain brought by synthetic data is obvious across all metrics. With the increasing of warm-up epochs, the final performance improves gradually. To verify the universality, the effect of warm-up on different datasets is presented in Figure 6. Unlike on CSL-Daily,

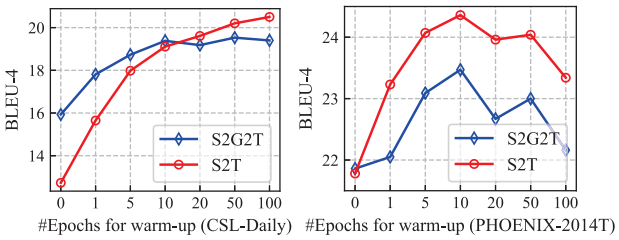

Figure 6. The effect of warm-up on different datasets.

Table 10. The quantity of synthetic data compared to parallel data participating in the training process on CSL-Daily.

	S2G2T				S2T			
quantity	R	B-2	B-3	B-4	R	B-2	B-3	B-4
0×	44.21	32.11	22.44	15.93	37.29	26.56	18.06	12.73
1×	45.62	33.84	23.98	17.30	40.66	29.97	21.03	15.24
5×	46.57	34.85	24.88	18.22	45.47	33.86	24.39	18.05
10×	47.13	35.42	25.28	18.50	46.85	35.08	25.80	19.43
>30×	**48.38**	**36.16**	**26.26**	**19.53**	**49.49**	**37.23**	**27.51**	**20.80**

Table 11. The quality of synthetic data on CSL-Daily. The number in (·) denotes the BLEU-4 score of T2G networks for SignBT.

	S2G2T				S2T			
	R	B-2	B-3	B-4	R	B-2	B-3	B-4
w/o synthetic	44.21	32.11	22.44	15.93	37.29	26.56	18.06	12.73
blank input	45.83	33.49	23.99	17.36	41.22	30.44	21.60	15.77
low (3.05)	46.31	34.41	24.78	18.21	43.78	31.76	22.85	16.91
medium (7.02)	47.64	35.56	25.77	19.08	46.15	33.96	24.66	18.50
High (11.63)	**48.38**	**36.16**	**26.26**	**19.53**	**49.49**	**37.23**	**27.51**	**20.80**

large warm-up epochs do not bring further improvement on PHOENIX-2014T but a slight decrease in the BLEU score. Considering that the topic of PHONIEX-2014T is all around weather forecast, it may constrain the learning of linguistic from synthetic data. Although we do collect some sentences about the weather, they account for a small slice of all data. Considering training time, we use 50 warm-up epochs for CSL-Daily and 10 for PHOENIX-2014T.

After warm-up, we use a small portion of synthetic data for training, which is sampled randomly after each epoch. In Table 9, we evaluate several mix ratios of training data. When synthetic data account for a small ratio, the S2T models get better performance, compared to models trained with directly giving them up. In contrast, the participation of synthetic data after warm-up consistently harms the S2G2T model. The noise comes mainly from the synthetic part. We argue that the noise in sparse gloss-level is difficult for the model to handle, while the noise in dense feature-level enables better generalization instead.

The Quantity and Quality of Synthetic Data. In Table 10, we train the SLT network with different quantity of synthetic data. The performance improves steadily when increasing synthetic data volume.

Besides, we analyse the performance variations using different quality of synthetic data. We train three text-to-gloss (T2G) networks with different epochs, *i.e.,* low, medium and high, whose BLEU-4 scores are 3.05, 7.02,

Table 12. Comparison with methods for SLT on PHOENIX-2014T (the higher the better).

	Dev					Test				
S2G2T	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4
SL-Luong [10]	44.14	42.88	30.30	23.02	18.40	43.80	43.29	30.39	22.82	18.13
SL-Transf. [11]	-	47.73	34.82	27.11	22.11	-	48.47	35.35	27.57	22.45
BN-TIN-Transf.^{2} (baseline)	47.83	47.72	34.78	26.94	21.86	47.98	47.74	35.27	27.59	22.54
BN-TIN-Transf.^{2}+BT (ours)	**49.53**	**49.33**	**36.43**	**28.66**	**23.51**	**49.35**	**48.55**	**36.13**	**28.47**	**23.51**
S2T	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4
SL-Luong [10]	31.80	31.87	19.11	13.16	9.94	31.80	32.24	19.03	12.83	9.58
Joint-SLRT [11]	-	47.26	34.40	27.05	22.38	-	46.61	33.73	26.19	21.32
TSPNet-Joint [25]	-	-	-	-	-	34.96	36.10	23.12	16.88	13.41
BN-TIN-Transf. (baseline)	46.87	46.90	33.98	26.49	21.78	46.98	47.57	34.64	26.78	21.68
BN-TIN-Transf.+SignBT (ours)	**50.29**	**51.11**	**37.90**	**29.80**	**24.45**	**49.54**	**50.80**	**37.75**	**29.72**	**24.32**
MCT [5]	45.90	-	-	-	19.51	43.57	-	-	-	18.51
STMC-T [54]	48.24	47.60	36.43	29.18	24.09	46.65	46.98	36.09	28.70	23.65

Table 13. Comparison with methods for SLT on CSL-Daily (the higher the better).

	Dev					Test				
S2G2T	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4
SL-Luong [10]	40.18	41.46	25.71	16.57	11.06	40.05	41.55	25.73	16.54	11.03
SL-Transf. [11]	44.18	46.82	32.22	22.49	15.94	44.81	47.09	32.49	22.61	16.24
BN-TIN-Transf.^{2} (baseline)	44.21	46.61	32.11	22.44	15.93	44.78	46.85	32.37	22.57	16.25
BN-TIN-Transf.^{2}+BT (ours)	**48.38**	**50.97**	**36.16**	**26.26**	**19.53**	**48.21**	**50.68**	**36.00**	**26.20**	**19.67**
S2T	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4
SL-Luong [10]	34.28	34.22	19.72	12.24	7.96	34.54	34.16	19.57	11.84	7.56
Joint-SLRT [11]	37.06	37.47	24.67	16.86	11.88	36.74	37.38	24.36	16.55	11.79
BN-TIN-Transf. (baseline)	37.29	40.66	26.56	18.06	12.73	37.67	40.74	26.96	18.48	13.19
BN-TIN-Transf.+SignBT (ours)	**49.49**	**51.46**	**37.23**	**27.51**	**20.80**	**49.31**	**51.42**	**37.26**	**27.76**	**21.34**

and 11.63, respectively. Those three T2G networks are then used to generate synthetic data of different qualities. As shown in Table 11, the metric scores of SLT model are higher with higher quality synthetic data. We also simulate the worst condition by pairing monolingual texts with blank input for training (corresponding to "blank input" in Table 11). While achieving a small gain, it has a performance gap compared to the models trained with synthetic data. It verifies that the synthetic pairs from our SignBT do make effects on the seq2seq learning, rather than simple enhancement in language modelling.

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

Our SignBT mechanism is dedicated to the S2T setting, which directly translates spoken language from videos. The results of back-translation on S2G2T are also provided.

Evaluation on PHOENIX-2014T: In Table 12, we compare our approach with SLT methods on PHOENIX-2014T. MCT [5] and STMC-T [54] are evaluated under multi-cue setting. TSPNet-Joint [25] explores gloss-free S2T methods with word-level sign language corpus for pre-training. Joint-SLRT [11] jointly models CSLR and SLT problems in one framework. Our baseline model is at the same level as previous methods. The SignBT approach gives an improvement of 2.6 BLEU-4 points on both sets.

Evaluation on CSL-Daily: In Table 13, we compare our approach with SLT methods on CSL-Daily. The performance boost with SignBT on CSL-Daily is more significant than that on PHOENIX-2014T, which is attributed to two factors. On one hand, Chinese sentences are built on three levels, *i.e.,* character, word, and sentence. The number of unique characters in sentences is 2.3K, but they have over 8K combinations in word-level. On the other hand, the vocabulary size of sign words in videos also exceeds 2K. The vocabulary sizes of both sides are quite large. When our SignBT approach serves as a data-augmentation method for the encoder-decoder framework, it is more effective when dealing with the large-scale vocabulary problem.

6. Conclusion

In this paper, we propose to improve the translation quality with monolingual data, which is rarely investigated in SLT. By designing a SignBT pipeline, we convert massive spoken language texts into source sign sequences. The synthetic pairs are treated as additional training data to alleviate the shortage of parallel data in training. With no change on the network architectures, our approach can be easily applied to encoder-decoder based SLT methods. Moreover, we contribute a large-scale SLT dataset with diverse topics and complete annotations. Extensive experiments demonstrate the significance of our sign back-translation approach.

Acknowledgements. This work was supported in part by the National Natural Science Foundation of China under Contract U20A20183, 61836011 and 62021001, and in part by the Youth Innovation Promotion Association CAS under Grant 2018497. It was also supported by the GPU cluster built by MCC Lab of Information Science and Technology Institution, USTC.

References

[1] Samuel Albanie, Gül Varol, Liliane Momeni, Triantafyllos Afouras, Joon Son Chung, Neil Fox, and Andrew Zisserman. BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues. *ECCV*, 2020. 1, 3

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2014. 2, 7

[3] Loïc Barrault, Ondřej Bojar, Marta R Costa-Jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, et al. Findings of the 2019 conference on machine translation (wmt19). In *WMT*, 2019. 1, 6

[4] Danielle Bragg, Oscar Koller, Mary Bellard, Larwan Berke, Patrick Boudreault, Annelies Braffort, Naomi Caselli, Matt Huenerfauth, Hernisa Kacorri, Tessa Verhoef, Christian Vogler, and Meredith Ringel Morris. Sign language recognition, generation, and translation: An interdisciplinary perspective. In *ACM ASSETS*, 2019. 1, 5

[5] Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. Multi-channel transformers for multi-articulatory sign language translation. In *ECCVW*, 2020. 2, 8

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017. 2, 6

[7] Ka Leong Cheng, Zhaoyang Yang, Qifeng Chen, and Yu-Wing Tai. Fully convolutional networks for continuous sign language recognition. *ECCV*, 2020. 2, 6

[8] Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Lip reading sentences in the wild. In *CVPR*, 2017. 2

[9] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden. Subunets: end-to-end hand shape and continuous sign language recognition. In *ICCV*, 2017. 6

[10] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. Neural sign language translation. In *CVPR*, 2018. 1, 2, 3, 4, 5, 8

[11] Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. Sign language transformers: Joint end-to-end sign language recognition and translation. In *CVPR*, 2020. 1, 2, 3, 4, 6, 8

[12] Runpeng Cui, Zhong Cao, Weishen Pan, Changshui Zhang, and Jianqiang Wang. Deep gesture video generation with learning on regions of interest. *TMM*, 22(10):2551–2563, 2020. 2

[13] Runpeng Cui, Hu Liu, and Changshui Zhang. Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. In *CVPR*, 2017. 2

[14] Runpeng Cui, Hu Liu, and Changshui Zhang. A deep neural framework for continuous sign language recognition by iterative training. *IEEE TMM*, 21(7):1880–1891, 2019. 2, 3, 6

[15] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*, 2006. 2, 3, 4

[16] Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. On using monolingual corpora in neural machine translation. *arXiv preprint arXiv:1503.03535*, 2015. 2

[17] Jie Huang, Wengang Zhou, Qilin Zhang, Houqiang Li, and Weiping Li. Video-based sign language recognition without temporal segmentation. In *AAAI*, 2018. 1, 3, 6

[18] Andrew J Hunt and Alan W Black. Unit selection in a concatenative speech synthesis system using a large speech database. In *ICASSP*, 1996. 2

[19] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3D convolutional neural networks for human action recognition. *IEEE TPAMI*, 35(1):221–231, 2013. 2

[20] Hamid Reza Vaezi Joze and Oscar Koller. Ms-asl: A large-scale data set and benchmark for understanding american sign language. *BMVC*, 2019. 3

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[22] Oscar Koller, 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 TPAMI*, 42(9):2306–2320, 2020. 1, 2

[23] Oscar Koller, Jens Forster, and Hermann Ney. Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. *CVIU*, 141:108–125, 2015. 3, 5

[24] Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *WACV*, 2020. 3

[25] Dongxu Li, Chenchen Xu, Xin Yu, Kaihao Zhang, Ben Swift, Hanna Suominen, and Hongdong Li. Tspnet: Hierarchical feature learning via temporal semantic pyramid for sign language translation. In *NeurIPS*, 2020. 1, 2, 3, 8

[26] Dongxu Li, Xin Yu, Chenchen Xu, Lars Petersson, and Hongdong Li. Transferring cross-domain knowledge for video sign language recognition. In *CVPR*, 2020. 1

[27] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *ACLW*, 2004. 6

[28] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *EMNLP*, 2015. 2, 7

[29] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? In *NeurIPS*, 2019. 6

[30] Zhe Niu and B. K. Mak. Stochastic fine-grained labeling of multi-state sign glosses for continuous sign language recognition. In *ECCV*, 2020. 2, 6

[31] Sylvie CW Ong and Surendra Ranganath. Automatic sign language analysis: A survey and the future beyond lexical meaning. *IEEE TPAMI*, 6:873–891, 2005. 2

[32] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002. 6

[33] Junfu Pu, Wengang Zhou, Hezhen Hu, and Houqiang Li. Boosting continuous sign language recognition via cross modality augmentation. In *ACM MM*, 2020. 2

[34] Junfu Pu, Wengang Zhou, and Houqiang Li. Iterative alignment network for continuous sign language recognition. In *CVPR*, 2019. 2

[35] Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. Progressive transformers for end-to-end sign language production. In *ECCV*, 2020. 2

[36] Rico Sennrich, Barry Haddow, and Alexandra Birch. Improving neural machine translation models with monolingual data. In *ACL*, 2016. 1, 2

[37] Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. Fingerspelling recognition in the wild with iterative visual attention. In *ICCV*, 2019. 1

[38] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. Animating arbitrary objects via deep motion transfer. In *CVPR*, 2019. 2

[39] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Neurips*. 2014. 2

[40] Advaith Sridhar, Rohith Gandhi Ganesan, Pratyush Kumar, and Mitesh Khapra. Include: A large scale dataset for indian sign language recognition. In *ACM MM*, 2020. 3

[41] T. Starner, J. Weaver, and A. Pentland. Real-time american sign language recognition using desk and wearable computer based video. *TPAMI*, 1998. 2

[42] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *NIPS*, 2014. 2

[43] Paul Taylor. *Text-to-speech synthesis*. Cambridge university press, 2009. 2

[44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2, 6

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 3, 6, 7

[46] Andrew Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE TIT*, 13(2):260–269, 1967. 4

[47] Ulrich Von Agris and Karl-Friedrich Kraiss. Towards a video corpus for signer-independent continuous sign language recognition. *Gesture in Human-Computer Interaction and Simulation, International Gesture Workshop*, 2007. 3

[48] Hanjie Wang, Xiujuan Chai, Xiaopeng Hong, Guoying Zhao, and Xilin Chen. Isolated sign language recognition with grassmann covariance matrices. *ACM TACCESS*, 8(4):1–21, 2016. 1, 3

[49] Shuo Wang, Dan Guo, Wen-gang Zhou, Zheng-Jun Zha, and Meng Wang. Connectionist temporal fusion for sign language translation. In *ACM MM*, 2018. 2

[50] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016. 4, 6

[51] Liang Xu, Hai Hu, Xuanwei Zhang, Lu Li, Chenjie Cao, Yudong Li, Yechen Xu, Kai Sun, Dian Yu, Cong Yu, Yin Tian, Qianqian Dong, Weitang Liu, Bo Shi, Yiming Cui, Junyi Li, Jun Zeng, Rongzhao Wang, Weijian Xie, Yanting Li, Yina Patterson, Zuoyu Tian, Yiwen Zhang, He Zhou, Shaoweihua Liu, Zhe Zhao, Qipeng Zhao, Cong Yue, Xinrui Zhang, Zhengliang Yang, Kyle Richardson, and Zhenzhong Lan. CLUE: A Chinese language understanding evaluation benchmark. In *COLING*, 2020. 5, 6

[52] Jihai Zhang, Wengang Zhou, Chao Xie, Junfu Pu, and Houqiang Li. Chinese sign language recognition with adaptive hmm. In *ICME*, 2016. 3

[53] Hao Zhou, Wengang Zhou, and Houqiang Li. Dynamic pseudo label decoding for continuous sign language recognition. In *ICME*, 2019. 4, 6

[54] Hao Zhou, Wengang Zhou, Yun Zhou, and Houqiang Li. Spatial-temporal multi-cue network for sign language recognition and translation. *IEEE TMM*, 2021. 2, 8