24th European Conference on Artificial Intelligence - ECAI 2020
Santiago de Compostela, Spain

Self-Attention-based Fully-Inception Networks for Continuous Sign Language Recognition

Mingjie Zhou [1] and **Michael Ng** [2] and **Zixin Cai** [3] and **Ka Chun Cheung** [4]

Abstract. In hearing-loss community, sign language is a primary tool to communicate with people while there is communication gap between hearing-loss people with normal hearing people. Continuous sign language recognition, which can bridge the communication gap, is a challenging task because of the weakly supervised ordered annotations where no frame-level label is provided. To overcome this problem, connectionist temporal classification (CTC) is the most widely used method. However, CTC learning could perform bad if extracted features are not good. For better feature extraction, this work presents the novel self-attention-based fully-inception (SAFI) networks for vision-based end-to-end continuous sign language recognition. Considering the length of sign words differs from each other, we introduce fully inception network with different receptive field to extract dynamic clip-level features. To further boost the performance, the fully inception network with an auxiliary classifier is trained with aggregation cross entropy (ACE) loss. Then the self-attention networks as global sequential feature extractor is used to model the clip-level features with CTC. The proposed model is optimized by jointly training with ACE on clip-level feature learning and CTC on global sequential feature learning in an end-to-end fashion. The best method in the baselines achieves 35.6% WER on validation set and 34.5% WER on test set. It employs a better decoding algorithm for pseudo label to do the EM-like optimization to fine tune CNN module. In contrast, our approach focuses on the better feature extraction for end-to-end learning. To alleviate the overfitting on the limited dataset, we employ temporal elastic deformation to triple the real-world dataset RWTH-PHOENIX-Weather 2014. Experimental results on the real-world dataset RWTH-PHOENIX-Weather 2014 demonstrate the effectiveness of our approach which achieves 31.7% WER on validation set and 31.3% WER on test set.

1 INTRODUCTION

Being one of the most significant methods of communication with hearing-loss people, sign language is used by millions of people in their daily life. However, the communication between hearing-loss and normal hearing community is inconvenient due to the language barrier. Thus, the sign language recognition (SLR) becomes meaningful in terms of bridging the communication gap between hearing-loss and normal hearing community.

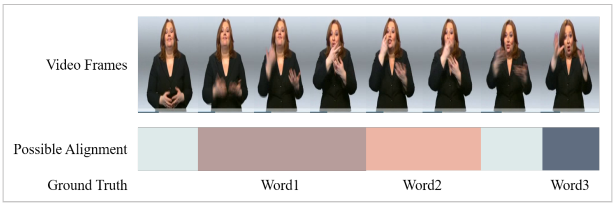


Figure 1: Illustration of weakly supervised CSLR problem where the ground truth is not in frame level. Besides, the video could contain noisy frames (the parts without label) which is not helpful for recognition.

Sign language recognition could be categorized into two types, i.e. isolated sign language recognition (ISLR) that recognizes segmented sign words one by one and continuous sign language recognition (CSLR) that recognizes a complete sentence of the sign language [11]. The isolated sign language recognition falls into disadvantage of the need of dramatic amount of human labor to segment the sign words from continuous sign videos or inaccurate temporal segmentation. Thus, continuous sign language recognition is more realistic. As for the CSLR, it is a weakly supervised problem where the ordered ground truth is not fine-grained provided as shown in Fig. 1. With the recent emergence of large scale dataset, the continuous sign language recognition becomes prevailing, e.g. [2, 17, 5, 14].

In this work, we focus on the CSLR. In terms of CSLR, the existing frameworks consist of two parts: feature extraction and sequence learning. For feature extraction, frame-level and clip-level features are widely used for sign language videos. Early work [15] designed fine-grained hand-crated features such as histogram of gradient (HOG), scale invariant feature transformation (SIFT), hand tracking patches and facial landmarks in frame level. Recent works [16, 18] show the superiority of deep features to hand-crafted features for CSLR. Among deep approaches, 2-D CNNs are the most used methods for extracting spatial features in frame level [2, 17]. However, 2-D CNNs only extract spatial feature without considering temporal dependencies, which is usually solved by RNNs. Besides, recognition in frame level could confront of redundant and noisy frames in the video which can affect optimization seriously. Thus, clip-level feature extraction is necessary for sign language videos to downsample frames and filter out noisy information. Some works [22, 29] apply sliding window to split video into clips and then feed them into 3-D CNN as clip-level features. Although 3-D CNN is more natural for spatiotemporal feature extraction, it is computationally expensive and suffering from tuning the huge number of parameters. Other work [4] extracts clip-level features by utilize 2-D and 1-D CNN separately, which obtains promising results with much less parameters to tune compared to 3-D CNN. However, the abovementioned clip-level

[1] Hong Kong Baptist University and NVIDIA AI Technology Center, Hong Kong SAR, email: 18481558@life.hkbu.edu.hk
[2] The University of Hong Kong, Hong Kong SAR, email: mng@maths.hku.hk
[3] Beijing Jiaotong University, China, email: 18120340@bjtu.edu.cn
[4] NVIDIA AI Technology Center, NVIDIA, Hong Kong SAR, email: chcheung@nvidia.com

feature extractors use a fixed receptive field for temporal feature extraction while the lengths of sign words could differ from each other. If the receptive filed could match the lengths of sign words, it could extract better features. Therefore, in this paper we consider (2+1)-D fully inception as the clip-level feature extractor which possesses dynamic receptive field and is helpful for sequence learning.

In terms of sequence learning, it aims at finding a mapping function from sequential features to the weakly supervised annotations. Inspired by the strong capacity of encoder-decoder framework for sequence-to-sequence problem, recent works [11, 22] consider it as a way to solve CSLR. In addition, the connectionist temporal classification (CTC) [7] is firstly designed for speech recognition which addresses ordered weakly supervised annotations the same as CSLR. SubUNets [2] is one of the first RNN-CTC methods in an end-to-end style to recognize German sign language. However, these methods may fail to achieve good performance due to the limited contribution of backpropagation to deep CNN feature extractor. The EM-like optimization is introduced [4, 29, 22], where E-step optimizes the whole sequence-to-sequence model and then generates fine-grained pseudo labels, and M-step fine tunes the CNN with pseudo labels for better features. In contrast, we introduce an end-to-end approach without EM-like iterations to achieve better performance.

In this paper, we present our self-attention-based fully-inception (SAFI) networks for the continuous sign language recognition and achieve the best performance among baselines. To extract spatiotemporal features better and efficiently, we introduce the (2+1)-D fully inception network with dynamic receptive field. In addition, an auxiliary classifier for clip-level feature learning with aggregation cross entropy (ACE) [28] boosts the performance. In terms of sequence learning, we employ the effective and efficient self-attention networks (SAN) [27] with CTC. Overall, the clip-level learning with ACE and sequence learning with CTC are jointly trained together. In short, the main contributions are listed as below:

- We develop an end-to-end architecture (SAFI) based on 2-D InceptionV1, i.e. GoogLeNet, 1-D inception modules and self-attention networks.
- To the best of our knowledge, we are the first to deploy (2+1)-D fully inception network for dynamic clip-level feature extraction and self-attention networks with CTC in order to obtain better performance on CSLR. In addition, we use ACE to improve the clip-level feature extraction.
- Experiments on RWTH-PHOENIX-Weather-2014, a large real-life continuous sign language dataset, demonstrate the effectiveness of our method.

The remainder of this paper is organized as follows. Section 2 lists the related work. Section 3 introduces our SAFI framework. A series of experiments is conducted and discussed in Section 4. Section 5 briefly draw the conclusion.

2 RELATED WORK

CSLR is a meaningful and challenging problem in real-world applications, which has attracted a lot of attention in the community of artificial intelligence and machine learning. Since sign language is one of the most efficient and widely used communication ways for the deaf-mute, CSLR is helpful to alleviate the communication gap between hearing-loss people with normal hearing people. A fundamental difficulty of CSLR is the hardness to capture visual semantics for videos according their target labels because of weakly supervised annotations problem.

To solve this problem, in literatures, researchers build frameworks which primarily consists of two modules, feature extractor and sequence learning. Early works employ hand-crafted features or 2-D CNN for frame-level features [15, 16, 18]. To match learned features with their target coarse-grained labels, hidden markov model (HMM) is applied. Inspired by the good performance of connectionist temporal classification (CTC), SubUNets [2] is proposed with three parts, i.e. cropped-hand video learning, full-frame video learning and combined learning. Specifically, the outputs of cropped-hand learning and full-frame learning are fed into CaffeNet and then modeled with BLSTM and learned with CTC respectively in frame level. In addition, the outputs of BLSTM in those two parts are fed to another BLSTM as combined learning with CTC. These methods have shown great potential for CSLR while they could confront of redundant and noisy frames in videos.

To emancipate the existing methods from noisy information, clip-level feature extractors are introduced in CSLR. Guo et al. [8] splits the video into several clips and proposes the two stream 2-D and 3-D CNNs with temporal convolutions as spatiotemporal feature extractor. Guo et al. [9] and Pu et al. [22] take the advantage of 3-D CNNs and encoder-decoder architectures to gain the comparable performance. Even though 3-D CNNs could have strong capacity of video representation, it requires more computational power and contains more parameters than 2-D CNNs, which makes it hard to train on the limited dataset. The following work introduces (2+1)-D CNNs to represent the video segments. Cui et al. [4] utilized GoogLeNet with temporal convolutions as clip-level feature extractor to filter out noisy frames and pretrained it with three-stage optimization. Firstly it generates alignment proposal through the end-to-end learning using VGG. Secondly, the generated alignment proposal helps fine tune the feature extractor using GoogLeNet in clip level by KL-divergence. Finally, the fine-tuned feature extractor along BLSTM is trained with CTC. Such three-stage optimization is similar to the EM-like optimization, which can lead to complex training process.

Besides, the above mentioned approaches base on RNNs to process the sequential features, while RNNs cannot be parallelized and may fall into local context. To learn global context and train the model in a higher speed, self-attention networks (SAN) is introduced with the famous neural machine model, i.e. transformer [27] which gives up the conventional sequence model, i.e. RNNs, and achieved the state-of-the-art results on both WMT2014 English German and English-French translation tasks.

Therefore, in this paper, we introduce the novel self-attention-based fully-inception networks to accurately extract features from sign videos and train it in an end-to-end style.

3 THE PROPOSED METHOD

In this section, we present the novel self-attention-based fully-inception (SAFI) networks for end-to-end continuous sign language recognition. Our method adopts (2+1)-D fully inception network and self-attention networks optimized jointly with clip-level feature learning and sequence learning.

3.1 Network architecture

The architecture of our method is shown in Fig. 2. The presented architecture consists of (2+1)-D fully inception network for clip-level feature extraction from the input video frames and self-attention networks for sequential feature extraction.

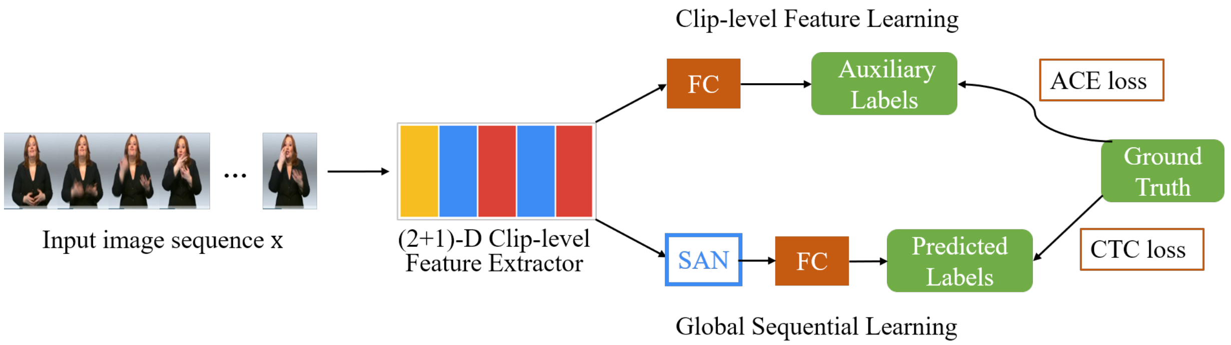

Clip-level Feature Learning
FC
Auxiliary Labels
ACE loss
Ground Truth
Input image sequence x
(2+1)-D Clip-level Feature Extractor
SAN
FC
Predicted Labels
CTC loss
Global Sequential Learning

Figure 2: The self-attention-based fully-inception (SAFI) networks with CTC loss and ACE loss which is trained by one-stage end-to-end learning for CSLR.

3.1.1 Clip-level feature extractor

Let $\mathbf{x} = \{x_t \in \mathbb{R}^{h \times w \times c}\}_{t=1}^{T}$ be the input sequence of a video with T frames. The clip-level spatiotemporal representation consists of 2-D InceptionV1 network, i.e. GoogLeNet [25], and stacked 1-D inception modules, denoted $\mathcal{F}_{\text{2D-Incep}}$ and $\mathcal{F}_{\text{1D-Incep}}$ respectively. With the success in ILSVRC 2014 [23], GoogLeNet [25] shows its discriminative power on spatial features. In terms of temporal feature learning, we consider the 1-D inception modules for its capacity of dynamic feature extraction. According to RWTH-PHOENIX-Weather-2012 dataset [6], the lengths of isolated signs could vary around 10 instead of a fix length. To aggregate local context of video frames and filter out noisy frames, it is suitable to apply convolutions with equivalent receptive field to the temporal domain. Considering this situation, we apply the stacked 1-D version of inception modules which contain kernels with different size, i.e. different receptive field. Therefore, the clip-level representation $\mathbf{r} = \{r_t \in \mathbb{R}^d\}_{t=1}^{\hat{T}}$ given the input video frames $\mathbf{x}$ is as:

$$\mathbf{r} = \mathcal{F}_{\text{1D-Incep}}(\mathcal{F}_{\text{2D-Incep}}(\mathbf{x})), \tag{1}$$

where $\hat{T} = T/4$ due to downsampling by pooling layers with zero padding on the time domain. The 1-D version of inception module consists of temporal kernels that have 1, 3, 5 kernel size respectively as shown in Fig. 3. With different size of kernels, the receptive field of this 1-D inception module ranges from 1 to 5. The proposed architecture contains two stacked layers of 1-D inception module so that the receptive field ranges from 4 to 16. Thanks to the capacity of spatiotemporal features, it is helpful for sequence learning with CTC.

3.1.2 Global sequential feature extractor

The self-attention networks (SAN) [27] works as global sequential feature extractor to compute the output states given the clip-level representation $\mathbf{r}$. The SAN allows to access sequential vectors at all time steps so that it would not fall into local context. Besides, the SAN can be parallelized due to the fully self-attention mechanism. Thus, the SAN is taken as our sequence model which gives the sequential vectors' representation as follows:

$$\mathbf{h} = \{h_t \in \mathbb{R}^d\}_{t=1}^{\hat{T}} = \mathcal{F}_{\text{SAN}}(\mathbf{r}), \tag{2}$$

where $\mathcal{F}_{\text{SAN}}$ denotes the transformation of self-attention networks. The self-attention networks consists of self-attention layers and feed-forward layers. To understand it easier, the self-attention layer is considered as the weighted summation of sequence vectors so that the representation at each time step contains information from global context. The feedforward layer is two linear transformations with ReLu activation in between for improving the capacity of SAN. Thanks to self-attention mechanism, SAN enables the parallelization and the access of global context at each time step.

3.2 Clip-level feature learning

Other than feeding clip-level feature to global sequential model directly, the presented architecture utilizes an auxiliary classifier for clip-level feature learning to boost the recognition performance. With the ability to classify the sign words in the clip-level, it helps the global sequential feature extraction. The auxiliary classifier is added after the clip-level feature extractor as below:

$$\tilde{\mathbf{y}} = \{\tilde{y}^t \in \mathbb{R}^K\}_{t=1}^{\hat{T}} = \{\text{softmax}(W_r r_t)\}_{t=1}^{\hat{T}}, \tag{3}$$

where $\tilde{\mathbf{y}}$ denotes the probability distributions over K words, and $W_r \in \mathbb{R}^{K \times d}$ is the linear projection matrix.

The auxiliary classifier enables the clip-level feature learning more directly instead of backpropagation from sequence learning. To make it possible, we employ the free-alignment aggregation cross entropy (ACE) [28] which is designed for scene text recognition. Different from the CTC (details in Section. 3.3), the aggregation cross entropy doesn't consider the alignment paths but to count the normalized numbers of words that emerge in a sequence:

$$\mathcal{L}_{\text{ACE}}(\mathbf{x}, \mathbf{z}) = -\sum_{k=1}^{|V|} \bar{\mathcal{N}}_k \log \bar{y}_k = -\sum_{k=1}^{|V|} \bar{\mathcal{N}}_k \log \sum_{t=1}^{\hat{T}} \tilde{y}_k^t / \hat{T} \tag{4}$$

where the normalized number of k-th sign word in the vocabulary is $\bar{\mathcal{N}}_k = \mathcal{N}_k / \hat{T}$, $\mathcal{N}_k$ is the number of the sign word k in the true sequential label $\mathbf{z}$, and $\tilde{y}_k^t$ denotes the output unit k at time t. In addition, the number of the *blank* symbol in the vocabulary is the difference of length of video clips and labels, i.e. $\mathcal{N}_{\textit{blank}} = \hat{T} - L$.

3.3 Global sequential feature learning

Given the extracted global sequential feature $\mathbf{h}$ from Eq. 2, the final linear projection can produce the prediction for the continuous sign language video:

$$\mathbf{y} = \{y^t \in \mathbb{R}^K\}_{t=1}^{\hat{T}} = \{\text{softmax}(W_h h_t)\}_{t=1}^{\hat{T}}, \tag{5}$$

where $\mathbf{y}$ denotes the probability distributions over K words, and $W_h \in \mathbb{R}^{K \times d}$ is the linear projection matrix.

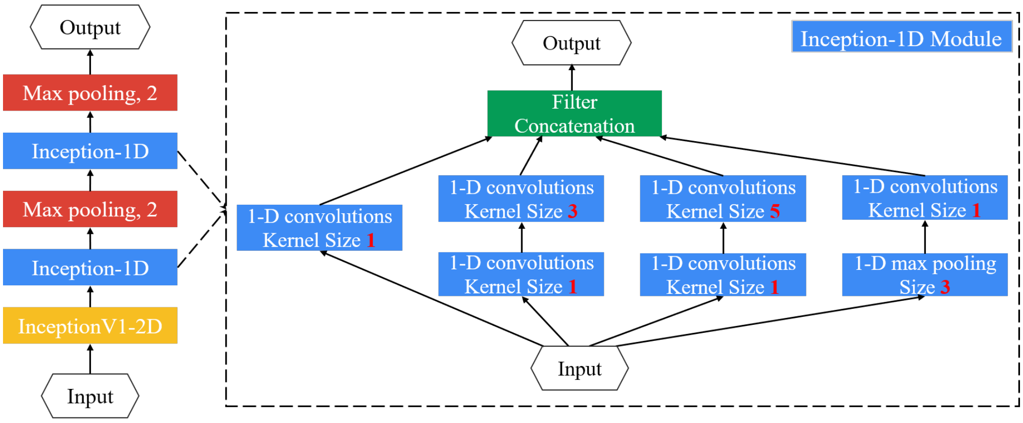

Output
Max pooling, 2
Inception-1D
Max pooling, 2
Inception-1D
InceptionV1-2D
Input
Inception-1D Module
Output
Filter Concatenation
1-D convolutions Kernel Size 1
1-D convolutions Kernel Size 3
1-D convolutions Kernel Size 5
1-D convolutions Kernel Size 1
1-D convolutions Kernel Size 1
1-D convolutions Kernel Size 1
1-D max pooling Size 3
Input

Figure 3: The illustration of clip-level feature extractor which contains stacked 1-D inception modules. The max pooling in red employs size 2 which downsamples the sequential features.

According to Eq. 5, y^t_k denotes the output unit k at time t. The output y^t represents the probability distribution over labels at time t and each unit, i.e. y^t_k, is interpreted as the probability of the k-th label at time t. Defines the probability over the vocabulary set V^T of length T sequences as follows:

$$p(\pi|\mathbf{x}) = \prod_{t=1}^{\hat{T}} y^t_{\pi_t}, \forall \pi \in V^T. \qquad (6)$$

Continuous sign language recognition is weakly supervised classification in which not every output at different time step is labelled. To solve this problem, we employ the connectionist temporal classification (CTC) [7] which considers the all possible alignments for the sequence outputs and the labels. Therefore, we could define the many-to-one mapping function $\mathcal{B}$ of a given labelling $\mathbf{z}$, e.g. $\mathcal{B}$(a-aab) $=$ $\mathcal{B}$(-aa-a-b) $=$ aab, where the '-' is the *blank* symbol in the vocabulary used to represent the empty clips and separate the repeated words, and the sum of probabilities of all possible alignment paths:

$$p(\mathbf{z}|\mathbf{x}) = \sum_{\pi \in \mathcal{B}^{-1}(\mathbf{z})} p(\pi|\mathbf{x}), \qquad (7)$$

which is calculated by HMM-like forward-backward algorithm for feasibility and further details will not be discussed here. The objective function of CTC is defined as:

$$\mathcal{L}_{\text{CTC}}(\mathbf{x}, \mathbf{z}) = -\log p(\mathbf{z}|\mathbf{x}) \qquad (8)$$

During training, the auxiliary ACE loss is taken as the regularization term in our final optimization. Besides, L_2 regularization is also considered. Let λ_1 and λ_2 be the tradeoff coefficient of ACE loss and L_2 regularization respectively. Thus, the final objective function of our model is:

$$\mathcal{L} = \frac{1}{|D|} \sum_{(\mathbf{x},\mathbf{z})\in D} (\mathcal{L}_{\text{CTC}}(\mathbf{x}, \mathbf{z}) + \lambda_1 \mathcal{L}_{\text{ACE}}(\mathbf{x}, \mathbf{z})) + \lambda_2 \|\theta\|_2^2, \qquad (9)$$

where θ is the weighting parameters of the model. Based on the Bayesian theory, the inferred labels $\hat{\mathbf{y}}$ can be formulated as follows,

$$\hat{\mathbf{y}} = \arg \max_{\mathbf{y}\in V^{\leq \hat{T}}} p(\mathbf{y}|\mathbf{x}).$$

During decoding stage, the labels are decoded by applying beam search algorithm [12] which only searches top-B possible candidates where the beam width B is a hyperparameter.

4 EXPERIMENTS

In this section, we will analyze the performance of our approach on the prevail continuous sign language dataset.

4.1 Experimental setup

4.1.1 Dataset

We evaluate our method on a real-life German sign language dataset, i.e. RWTH-PHOENIX-Weather 2014 [15] which is a prevail benchmark for continuous sign language task. To our best knowledge, this is the largest public real-life dataset for sign language recognition. The dataset contains both full-frame and cropped-hand videos of signers. The dataset is split to training set (5672 sentences, 10 hours), development set(540 sentences, 0.84 hours) and test set(629 sentences, 0.99 hours) by the release. Note that there exist words which never occur in training set called out of vocabulary. The size of vocabulary is 1232 including 1231 unique words in training set and a *blank* symbol for CTC.

4.1.2 Evaluation metric

To evaluate our method and compare with other methods, we take word error rate (WER) as the quantitative indicator which is the most widely used in continuous sign language recognition task [4, 8, 22, 29]. Word error rate bases on edit distance which counts the minimum number of operations, i.e. insertions (ins), deletions (del) and substitutions (sub), required to transform inferred sequence to the ground truth:

$$\text{WER} = \frac{\text{\# insertions} + \text{\# deletions} + \text{\# substitutions}}{\text{\# words in ground truth}}. \qquad (10)$$

Lower value of WER indicates better performance of continuous sign language recognition.

4.1.3 Baselines

We aims at improving continuous sign language recognition in an end-to-end style. The following representative methods, which are widely use to tackle CSLR, are taken as our baselines.

1-Mio-H+CMLLR [15, 16] is an approach that demonstrates the superiority of deep features to previous hand-crafted features [15]. In [16], the 1-million-hands model is built to classify the hand shapes of cropped-hand video on frame level. The 1-million-hands model bases on the HMM with EM algorithm where the visual model is replaced by GoogLeNet instead of gaussian mixture model (GMM) in [15]. In our method, we don't introduce complementary information from 1-million-hands dataset.

Hybrid CNN-HMM [18] embeds the GoogLeNet into hybrid HMM which is popular in automatic speech recognition. This method takes cropped-hand video as inputs, utilize the best alignment results generated by **1-Mio-H+CMLLR** and then train the CNN-HMM in an end-to-end fashion.

SubUNets [2] is one of the first end-to-end frameworks proposed for CSLR. SubUNets takes advantage of cropped-hand and full-frame videos as two-stream inputs. It consists of two BLSTM-CTC parts respectively on two-stream inputs and an extra BLSTM-CTC part on both BLSTM layers. During testing, SubUNets utilize a HMM-like topology with language model to decode the inference.

Staged-Opt [4] takes the advantage of three-stage optimization. VGG and GoogLeNet with stacked temporal convolutions is taken as feature extractor, and BLSTM with CTC is used for end-to-end training. Staged-Opt firstly generates alignment proposal through the end-to-end learning using VGG. Secondly, the generated alignment proposal helps fine tune the feature extractor using GoogLeNet in clip level. Finally, the fine-tuned feature extractor along BLSTM is trained with CTC.

2D&3D-CNN+Stacked-1D-CNN [8] takes the advantage of two stream CNN for CSLR. The 2-D and 3-D ResNets can extract features in frame level and clip level respectively for recognition. After extracting 2-D spatial features, it is fed to four-layer temporal convolutions to obtain spatiotemporal features. Then the spatiotemporal features extracted from 2-D and 3-D ResNets are fused together. Finally, the fused features modeled by BGRU are trained jointly with cross entropy, CTC and triplet loss in an end-to-end fashion.

3D-CNN+EncDec-CTC-DTW [22] focuses on learning for weakly supervised data in different ways. The video is segmented to several clips as the inputs of 3-D ResNets to extract spatiotemporal features. This baseline employs the encoder-decoder architecture with the CTC and soft-DTW imposed on encoder and decoder respectively. The EM-like method is employed to fine tune the 3-D ResNets for optimization.

3D-CNN+TEM+CTC [29] mainly improves EM-like optimization to achieve better recognition performance. The dynamic pseudo label decoding is proposed to generate better pseudo label during iteration so that the training could have a correct direction. The network architecture in [29] consists of Inception-3D [1], the BGRU and temporal convolutions together. The whole architecture is trained with CTC and EM-like optimization based on the dynamic pseudo label decoding.

4.1.4 Implementation details

In our experiments, the input video frames are resized to 224 × 224 for the GoogLeNet. All input frames are the full-frame data of RWTH-PHOENIX-Weather 2014 instead of hand-tracked data. The GoogLeNet [25] is initialized with weights pretrained on ILSVRC-2014 [23]. The rest of parameters are initialized by he-normal [10] initializer which is designed for networks with ReLu activation. We take the output of *Mix_5c* layer in GoogLeNet as the input of 1-D inception modules. The spatial dropout probability is set to 0.3. For the stacked 1-D inception modules, we maintain the same settings of *Mix_5c* layer in GoogLeNet. Both two-layer inception modules share the same hyperparameters as shown in Tab. 1. Therefore, the outputs of inception module would be the sequence vectors with 1024 dimension. For the self-attention networks, we employ the 1-layer encoder of Transformer-big model as described in [27] but with filter size 2048 in the feedforward network. We choose $\lambda_1 = 2.5$ and $\lambda_2 = 4 \times 10^{-5}$ as the weights of ACE loss and L_2 regularization respectively. We adopt ADAM [13] as the stochastic optimization approach with a fixed learning rate to 5×10^{-5}, batch size to 1. Our model is trained for 250,000 iterations and we evaluate our model every 3,000 iterations. During inference, we set the beam width of beam search to 10.

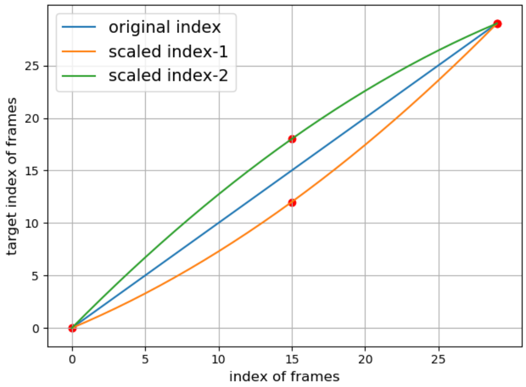


Figure 4: A illustration of how the elastic temporal scaling works for changing speed of a video. The target index of frames is generated from the transformation of second order polynomial that goes through $(0, 0)$, $(\frac{T}{2}, \frac{T}{2} \pm \mu T)$ and (T, T), i.e. two curves index-1 and index-2 respectively.

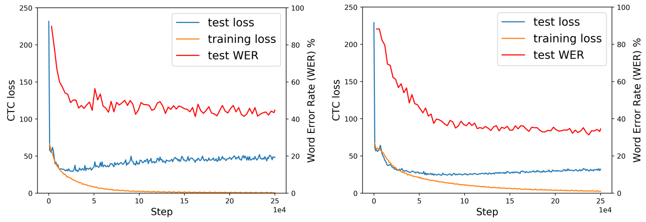
Table 1: The hyperparameters of 1-D inception module.

Branch	Kernel Size	# of filters
0	1	384
1	1	192
1	3	384
2	1	48
2	5	128
3	3	128

4.2 Overfitting reduction

At presents, the biggest real-life continuous sign language dataset for recognition problem is RWTH-PHOENIX-Weather multi-signer 2014 [15]. However, 10-hour training set is not big enough to train a deep neural network with millions of parameters. Therefore, overfitting becomes the nightmare that limits the performance of our architecture.

Data augmentation is one of the simplest and most common method to diminish overfitting [19]. Some image transformation methods discussed in [3, 21] are not effective to reduce overfitting on CSLR since the continuous movement pattern plays a more significant role in recognizing sign language. The speed of unseen gestures may differ from that of the training set. To improve the generalization of our architecture, we triple the RWTH-PHOENIX-Weather multi-signer 2014 dataset by applying temporal elastic deformation (TED) [20] in a way of changing the speed of videos. Different from



(a) without overfitting reduction (b) with overfitting reduction

Figure 5: The CTC loss and WER on RWTH-PHOENIX-Weather 2014

inserting or dropping some frames directly, the elastic temporal scaling can change the speed of the videos but keep the same length. Temporal elastic deformation makes use of second order polynomial curve fitting to change the position of video frames as shown in Fig. 4. Denote the index of time step in a video by x, the target index by y, a control point by (m, n) and the video length by T. The temporal scaling function then is given by second order polynomial curve fitting which goes through three coordinates $(0, 0)$, (m, n) and (T, T). In practice, the control point (m, n) is selected as $(\frac{T}{2}, \frac{T}{2} \pm \mu T)$, where μ is the coefficient that controls the deformation. Different control points with $\mu \in \{0.05, 0.10, 0.15, 0.20\}$ are tested to show how the temporal data augmentation affects recognition performance as shown in Tab. 3. Except for TED, random cropping is applied to each video during training.

In addition, dropout [24] has been a very effective tool to prevent overfitting. Considering the limited dataset and similarity of video frames, we apply spatial dropout [26] before low-level layers *Mix_3b* and *Mix_4b* of GoogLeNet. The dataset contains only 9 signers with the fixed background. Therefore, the GoogLeNet could get overfitting easily on the similar frames. The spatial dropout works on the feature channel of convolutions so that it could dropout some features during training to alleviate overfitting.

As shown in Fig. 5, contrast experiment is made to demonstrate the effectiveness of overfitting reduction. In contrast to that without overfitting reduction, the CTC loss on test set will barely increase after it reaches the lowest point if the model is trained with overfitting reduction. Besides, overfitting reduction leads to a lower word error rate.

4.3 Experimental results

Table 2: The ablation study of 1-D inception module and clip-level learning with ACE on RWTH-PHOENIX-Weather 2014.

Methods	VAL(%)		TEST(%)	
	del/ins	**WER**	**del/ins**	**WER**
InceptionV1+Conv1D-3	13.4/3.4	33.9	12.4/3.5	33.5
Fully-Inception	16.6/1.7	33.0	15.2/1.9	32.3
Fully-Inception+ACE	16.6/1.8	**31.7**	15.1/1.7	**31.3**

4.3.1 Effect of beam width

Beam search algorithm [12] is one of the most common methods to decode inference for CTC. The beam search takes top-B candidates during decoding at each time step since searching for all possible results is exponential and unrealistic. The computational complexity of beam search is $O(T \cdot B \cdot K \log(B \cdot K))$ where T is the length of input sequence, B is beam width and K is the size of vocabulary. Larger beam width can indicate better recognition result but more time for decoding. Usually beam search can work well but we need to figure out what beam width results in balanced performance on recognition accuracy and speed. Therefore, we apply parameter search to find the optimal beam width on our approach. As shown in Fig. 6, there is only small improvement on WER when beam width is larger than 3. Consider the balance between WER and speed, we choose beam width to 10 which produces good WER on both validation and test set.

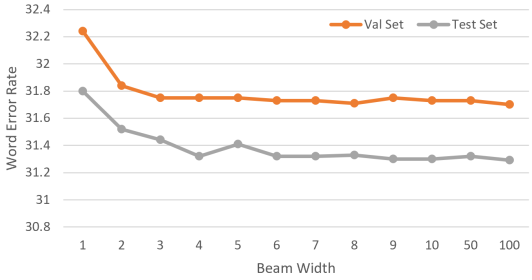


Figure 6: The effect of beam width on SAFI performance.

4.3.2 Ablation study

This part investigates the effectiveness of our modules, i.e. 1-D inception module and clip-level learning with ACE loss. In the Tab. 2, the ”InceptionV1+Conv1D-3” represents the training without ACE, and the 1-D inception modules of the clip-level feature extractor are replaced by 1-D convolutions with kernel size 3 and 1024 filters. The ”Fully-Inception” represents the training without clip-level learning with ACE. The ”Fully-Inception+ACE” represents the full version of our proposed method. To compare it fairly, the rest parts of these settings are the same. By comparing ”InceptionV1+Conv1D-3” with ”Fully-Inception”, the 1-D inception module contributes 0.9% WER improvement from 33.9% to 33.0% on validation set and 1.2% WER improvement from 33.5% to 32.3% on testing set. In addition, the clip-level learning with ACE further boosts the performance to 31.7% WER on validation set and 31.3% WER on testing set. Therefore, the fully inception feature extractor and clip-level feature learning with ACE contribute to the performance obviously.

4.3.3 Alignment and comparisons

Fig. 7 shows an example of self-attention heatmap and alignment for our architecture from test set. The self-attention heatmap indicates the correlation between clip queries and keys. According to Fig. 7, the alignment result is related to the self-attention heatmap where correlations, to some extent, indicate the period of appearance of words.

In Tab. 3, we evaluate our method with the baselines on RWTH-PHOENIX-Weather multi-signer 2014 dataset by WER[5]. From the Tab. 3, the control point with $\mu = 0.15$ as described in Section 4.2 allows our approach to achieve the best recognition performance. The quantitative results demonstrate the effectiveness of our method on

[5] We only list deletions, insertions and WER since WER is the summation of deletions, insertions and substitutions.

Table 3: WER of different methods on the RWTH-PHOENIX-Weather 2014. The ”Iterations” shows how many the EM-like iterations are needed for optimization. ”-” means end-to-end training with no iteration needed. ”Hand”, ”Traj” (Trajectory) and ”Face” are the different input sources extracted from the original full-frame dataset. Among the baselines, the methods proposed in [16, 18] take advantage of extra supervision.

Methods	Iterations	**Input Source**	**VAL(%)**		**TEST(%)**	
			del/ins	**WER**	**del/ins**	**WER**
1-Mio-H+CMLLR [15, 16]	3	Hand, Traj, Face	16.3/4.6	47.1	15.2/4.6	45.1
SubUNets [2]	-	Full Frame, Hand	14.6/4.0	40.8	14.3/4.0	40.7
Hybrid CNN-HMM [18]	3	Hand	12.6/5.1	38.3	11.1/5.7	38.8
Staged-Opt-init [4]	-	Hand	16.3/6.7	46.2	15.1/7.4	46.9
Staged-Opt [4]	3	Hand	13.7/7.3	39.4	12.2/7.5	38.7
2D&3D-CNN+Stacked-1D-CNN [8]	-	Full Frame	11.6/6.3	38.9	10.9/6.4	38.7
3D-CNN+EncDec-CTC-DTW [22]	4	Full Frame	12.9/2.6	37.1	13.0/2.5	36.7
3D-CNN+TEM+CTC [29]	5	Full Frame	9.5/3.2	35.6	9.3/3.1	34.5
Ours(SAFI, $\mu = 0.05$)	-	Full Frame	16.7/1.5	32.9	16.0/1.7	32.9
Ours(SAFI, $\mu = 0.10$)	-	Full Frame	15.2/2.1	32.1	13.9/2.1	32.2
Ours(SAFI, $\mu = 0.15$)	-	Full Frame	16.6/1.8	**31.7**	15.1/1.7	**31.3**
Ours(SAFI, $\mu = 0.20$)	-	Full Frame	19.1/1.4	33.3	17.1/1.5	32.2

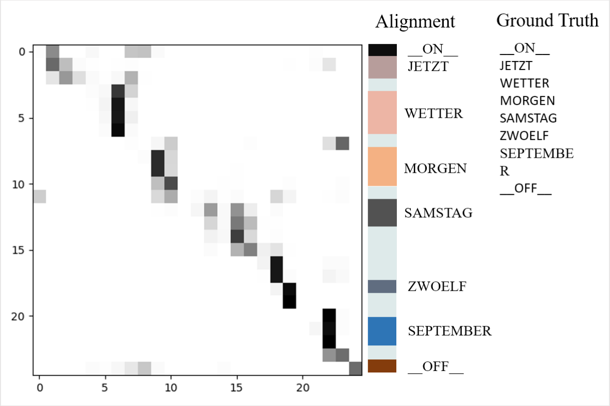


Figure 7: An example of self-attention heatmap and alignment between a video and its annotations. The x-axis is the clip index of queries in SAN, and the y-axis is the clip index of keys in SAN. The alignment parts without labels mean *blank* symbols in CTC.

continuous sign language recognition. Besides, the Fig. 7 indicates that our approach benefits from clip-level features and the SAN.

Among the baselines, 1-Mio-H+CMLLR, Hybrid CNN-HMM and SubUNets address sign video stream in frame level. The frame-level features could contain redundant and noisy information which can degrade the recognition performance. Staged-Opt extracts features in clip level by applying temporal convolutions and pooling layers to downsample sequential vectors. In addition, other approaches [8, 22, 29] split a video into several clips and feed them into 3-D CNNs as clip-level features while 3-D CNNs are computationally expensive and contain more parameters, e.g. 3-D ResNet-18 takes about 33.2M parameters while 2-D InceptionV1 takes 5˜6M parameters. These methods doesn’t consider the dynamic of length of sign words. In contrast, our approach employs (2+1)-D fully inception network to capture the dynamic temporal features to achieve better performance. Different from the abovementioned methods, our architecture is RNN-free network thanks to the self-attention networks (SAN) which makes training much faster.

Some approaches, i.e. 1-Mio-H+CMLLR, Hybrid CNN-HMM and SubUNets, apply multiple input sources. Besides, 1-Mio-H+CMLLR and Hybrid CNN-HMM take extra supervision from 1-million-hands dataset. On the contrary, our method only take the original full-frame video as input without extra supervision, which shows the strong capacity of our proposed method. In addition, many approaches introduce EM-like iterative optimization [15, 16, 18, 4, 22, 29] to optimize their CNN module to get the comparable performance. Compared to Staged-Opt-init, the improvement of Staged-Opt demonstrates the power of EM-like iterative optimization. In contrast, our method is end-to-end learning without any iteration. Overall, our competitive results benefit from better feature extraction and the strong capacity of our approach. We employ fully inception network to extract dynamic temporal features and utilize SAN to learn global context instead of using RNNs. The Fig. 7 indicates the effectiveness of SAN on CSLR. Furthermore, we boosts the performance by adding an auxiliary classifier learned with ACE. Therefore, our approach is able to achieve promising performance.

5 CONCLUSION

In this paper, we present a novel end-to-end approach for continuous sign language recognition. Our approach takes the advantage of fully inception, self-attention networks with CTC and clip-level feature learning with ACE. The stacked fully inception modules are capable of extracting dynamic local temporal features and filter out the noisy information. The self-attention networks with CTC can extract global sequential features effectively. Furthermore, the clip-level feature learning with ACE boosts the recognition performance. Experiments on the largest real-life dataset RWTH-PHOENIX-Weather multi-signer 2014 demonstrate the effectiveness of our approach.

ACKNOWLEDGEMENTS

This work is done during internship at NVIDIA AI Technology Center (NVAITC). This work is supported in part by the HKRGC GRF 12306616, 12200317, 12300218 and 12300519.

REFERENCES

[1] Joao Carreira and Andrew Zisserman, 'Quo vadis, action recognition? a new model and the kinetics dataset', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (July 2017).

[2] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden, 'Subunets: End-to-end hand shape and continuous sign language recognition', in *The IEEE International Conference on Computer Vision (ICCV)*, (Oct 2017).

[3] Dan Cireşan, Ueli Meier, and Juergen Schmidhuber, 'Multi-column Deep Neural Networks for Image Classification', *arXiv:1202.2745 [cs]*, (February 2012).

[4] Runpeng Cui, Hu Liu, and Changshui Zhang, 'Recurrent convolutional neural networks for continuous sign language recognition by staged optimization', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (July 2017).

[5] 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, (July 2019).

[6] Jens Forster, Christoph Schmidt, Thomas Hoyoux, Oscar Koller, Uwe Zelle, Justus Piater, and Hermann Ney, 'RWTH-PHOENIX-weather: A large vocabulary sign language recognition and translation corpus', in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pp. 3785–3789, Istanbul, Turkey, (May 2012). European Language Resources Association (ELRA).

[7] 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*, pp. 369–376, New York, NY, USA, (2006). ACM.

[8] Dan Guo, Shengeng Tang, and Meng Wang, 'Connectionist Temporal Modeling of Video and Language: a Joint Model for Translation and Sign Labeling', in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 751–757, Macao, China, (August 2019).

[9] Dan Guo, Wengang Zhou, Houqiang Li, and Meng Wang, 'Hierarchical lstm for sign language translation', in *AAAI Conference on Artificial Intelligence*, (2018).

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, 'Delving deep into rectifiers: Surpassing human-level performance on imagenet classification', in *The IEEE International Conference on Computer Vision (ICCV)*, (December 2015).

[11] Jie Huang, Wengang Zhou, Qilin Zhang, Houqiang Li, and Weiping Li, 'Video-based sign language recognition without temporal segmentation', in *AAAI Conference on Artificial Intelligence*, (2018).

[12] K. Hwang and W. Sung, 'Character-level incremental speech recognition with recurrent neural networks', in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5335–5339, (March 2016).

[13] Diederik P. Kingma and Jimmy Ba, 'Adam: A method for stochastic optimization', in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, (2015).

[14] 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 Transactions on Pattern Analysis and Machine Intelligence*, (2019).

[15] 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, (December 2015).

[16] Oscar Koller, Hermann Ney, and Richard Bowden, 'Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (June 2016).

[17] Oscar Koller, Sepehr Zargaran, and Hermann Ney, 'Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (July 2017).

[18] Oscar Koller, Sepehr Zargaran, Hermann Ney, and Richard Bowden, 'Deep Sign: Hybrid CNN-HMM for Continuous Sign Language Recognition', in *Procedings of the British Machine Vision Conference 2016*, York, UK, (2016).

[19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, 'ImageNet Classification with Deep Convolutional Neural Networks', in *Advances in Neural Information Processing Systems 25*. Curran Associates, Inc., (2012).

[20] Pavlo Molchanov, Shalini Gupta, Kihwan Kim, and Jan Kautz, 'Hand gesture recognition with 3d convolutional neural networks', in *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Boston, MA, USA, (June 2015). IEEE.

[21] Luis Perez and Jason Wang, 'The Effectiveness of Data Augmentation in Image Classification using Deep Learning', *arXiv:1712.04621 [cs]*, (December 2017).

[22] Junfu Pu, Wengang Zhou, and Houqiang Li, 'Iterative alignment network for continuous sign language recognition', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (June 2019).

[23] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, 'ImageNet Large Scale Visual Recognition Challenge', *International Journal of Computer Vision (IJCV)*, **115**(3), 211–252, (2015).

[24] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, 'Dropout: A simple way to prevent neural networks from overfitting', *Journal of Machine Learning Research*, **15**, 1929–1958, (2014).

[25] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, 'Going deeper with convolutions', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (June 2015).

[26] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler, 'Efficient object localization using convolutional networks', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (June 2015).

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin, 'Attention is all you need', in *Advances in Neural Information Processing Systems 30*, 5998–6008, Curran Associates, Inc., (2017).

[28] Zecheng Xie, Yaoxiong Huang, Yuanzhi Zhu, Lianwen Jin, Yuliang Liu, and Lele Xie, 'Aggregation cross-entropy for sequence recognition', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (June 2019).

[29] H. Zhou, W. Zhou, and H. Li, 'Dynamic pseudo label decoding for continuous sign language recognition', in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1282–1287, (July 2019).