Dilated Convolutional Network with Iterative Optimization for Continuous Sign Language Recognition

Junfu Pu, Wengang Zhou, Houqiang Li
CAS Key Laboratory of Technology in Geo-spatial Information Processing and Application System,
EEIS Department, University of Science and Technology of China
pjh@mail.ustc.edu.cn, zhwg@ustc.edu.cn, lihq@ustc.edu.cn

Abstract

This paper presents a novel deep neural architecture with iterative optimization strategy for real-world continuous sign language recognition. Generally, a continuous sign language recognition system consists of visual input encoder for feature extraction and a sequence learning model to learn the correspondence between the input sequence and the output sentence-level labels. We use a 3D residual convolutional network (3D-ResNet) to extract visual features. After that, a stacked dilated convolutional network with Connectionist Temporal Classification (CTC) is applied for learning the mapping between the sequential features and the text sentence. The deep network is hard to train since the CTC loss has limited contribution to early CNN parameters. To alleviate this problem, we design an iterative optimization strategy to train our architecture. We generate pseudo-labels for video clips from sequence learning model with CTC, and finetune the 3D-ResNet with the supervision of pseudo-labels for a better feature representation. We alternately optimize feature extractor and sequence learning model with iterative steps. Experimental results on RWTH-PHOENIX-Weather, a large real-world continuous sign language recognition benchmark, demonstrate the advantages and effectiveness of our proposed method.

1 Introduction

Sign language is one of the most efficient and widely used communication ways for the deaf-mute. It conveys sematic meaning through gestures, hand motions, even facial expressions, and so on. This makes the sign language a perfect test bed for computer vision, natural language processing, and human-computer interaction. The target of sign language recognition (SLR) is to automatically translate the sign videos into text or interpret it into spoken language. With broad social impact, it has attracted considerable attention. [Koller *et al.*, 2015; Camgoz *et al.*, 2017; Pu *et al.*, 2016; Cui *et al.*, 2017; Guo *et al.*, 2018]

Sign language recognition tasks are usually divided into two categories, *i.e.*, isolated SLR and continuous SLR. The main difference between these two tasks is the supervision information. The former is a kind of action classification, while the later not only recognizes the whole sign words but in correct order as well. Hence, continuous sign language recognition is much more complicated than isolate sign language recognition in general. The main idea of continuous sign language recognition is to learn the corresponding relationships between the input visual frames and the supervision of sentence-level sequential labels.

Continuous sign language recognition (CSLR) is somehow a kind of weakly supervised sequence learning task. Generally speaking, the performance of continuous SLR system usually depends on two aspects. On one hand, since the system takes visual sequences such as videos and images as inputs, it is significant to extract descriptive and discriminative representation of the visual inputs. Hence, the feature extractor with more representative capacity could result in a better performance. On the other hand, as a weekly supervised sequence learning task, it requires an accurate alignment between the input sequences and the sentence-level labels.

Recently, deep learning methods achieve breakthroughs in computer vision. Deep neural features outperform handcrafted features in most of computer vision tasks. Recent successes of Residual Networks (ResNet) [He *et al.*, 2016] in various image classification tasks prove that ResNet do have better representation capacities for images than other deep architectures. Meanwhile, 3D convolutional neural networks also demonstrate good performances in action recognition [Ji *et al.*, 2013; Tran *et al.*, 2015; Qiu *et al.*, 2017]. Inspired by the superiorities of Residual Networks and 3D neural networks, we use a combined 3D residual convolutional neural network for feature extraction in our continuous sign language recognition model.

In addition, many sequence learning methods achieve state-of-the-art performances in machine translation, speech recognition, and video caption [Song *et al.*, 2017; Li *et al.*, 2017]. As one of the representative sequence learning model, Long Short-Term Memory (LSTM) shows powerful capability to deal with sequential modelling tasks. However, there still remains some weakness. For instance, it's difficult to deal with long-range temporal dependencies, and it converges with a low speed in training. Oord *et al.* propose a dilated causal convolution-based architecture called WaveNet [Oord *et al.*, 2016a] for audio generation. It overcomes the prob-

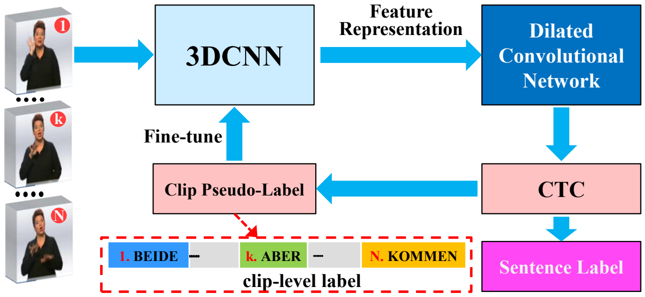


Figure 1: Iterative training illustration. 3D-CNN transpose the input clips into the fixed-length features for stacked dilated convolutions with CTC. The sequence learning model generates clip-level pseudo-labels to fine-tune the 3D-CNN in next iteration.

lem of long-range temporal dependencies and performs better than LSTM. Considering the similarity between SLR and audio generation task, dilated convolutional network has great potential for continuous sign language recognition. In this work, we use dilated convolutions with Connectionist Temporal Classification (CTC) loss to model the dependencies between different sign words. The CTC approach is originally proposed in [Graves *et al.*, 2006] for end-to-end speech recognition, and it has achieved significant improvement in speech recognition tasks.

In this paper, we propose a novel architecture for continuous sign language recognition and achieve state-of-the-art performance. It's worthwhile to highlight our main contributions as follows:

- We develop our architecture based on 3D residual network and dilated convolutions, which is a fresh framework for continuous sign language recognition task. To the best of our knowledge, we are the first to deploy dilated convolutions for sequence learning in continuous SLR system.
- We propose an iterative optimization strategy with Connectionist Temporal Classification (CTC) for our sign language recognition system (shown in Figure 1).
- Experiments on RWTH-PHOENIX-Weather, a large real-world continuous SLR benchmark, demonstrate the effectiveness and superiority of our method.

The rest of this paper is organized as follows: we first discuss some related works in Section 2. After that, Section 3 describes our proposed continuous sign language recognition framework and iterative optimization strategy in details. In Section 4, we conduct a series of experiments on a large benchmark to demonstrate the advantages of the proposed approach, and analyze how the iterative optimization strategy makes the performance better step by step. Finally, we make concluding remarks in Section 5.

2 Related Work

This section reviews the existing sign language recognition methods with different features and architectures. We also discuss some other sequence learning tasks related to the techniques used in our proposed approach.

We briefly group the methods for sign language or gesture recognition into two categories: hand-crafted feature based and deep learning based methods. Early works [Starner *et al.*, 1998; Wang *et al.*, 2006; Koller *et al.*, 2015] mostly use hand-crafted features with sequence modelling architectures like Hidden Markov Models (HMM) or Hidden Conditional Random Fields (HCRF). One typical work [Starner *et al.*, 1998] presents a real-time Hidden Markov Model-based system for recognizing sentence-level continuous American Sign Language (ASL) using a single camera to track the users' unadorned hands. Although experiments on a 40-word lexicon dataset show the efficiency of HMMs for sign language recognition, challenges still remain such as it's difficult to accommodate long-range dependencies among observations. To address this problem, Wang et al. [Wang *et al.*, 2006] derive a discriminative sequence model with Hidden Conditional Random Field (HCRF) for gesture recognition. The proposed model extends previous models for spatial CRFs into the temporal domain.

Previous works for continuous SLR generally conduct experiments on small datasets with a small vocabulary due to the lack of sign data. To alleviate the problem, Koller *et al.* [Koller *et al.*, 2015] release a large vocabulary real-life continuous sign language recognition dataset (RWTH-PHOENIX-Weather) which contains more than 1000 sign words. Recently, benefitting from the development of deep learning, sign language recognition has made great progress. The large datasets such as RWTH-PHOENIX-Weather make it possible to train deep neural networks for sign language recognition. The architecture for continuous sign language recognition usually consists of a visual input encoder for feature extraction, and a sequence learning model for mining the correspondence between the input sequence and sentence-level labels. There are several kinds of feature extraction encoders for sign language or gesture recognition, such as convolutional neural networks (CNNs) [Koller *et al.*, 2016a], 3D-CNNs [Huang *et al.*, 2015; 2018], and temporal convolutions [Pigou *et al.*, 2015].

There are many techniques for sequence learning. The most popular one is Recurrent Neural Network (RNN), including Long Short-Term Memory (LSTM) [Hochreiter and Schmidhuber, 1997] and Gated Recurrent Unit (GRU) [Cho *et al.*, 2014]. LSTM has been successfully used for many applications such as video captioning and question-answering [Kim *et al.*, 2017; Song *et al.*, 2017; Zhao *et al.*, 2017]. Even though LSTM demonstrates powerful capacity for sequence modelling, there still exist some nontrivial issues. While implementing the Back-Propagation Through Time (BPTT) of LSTM, the calculations of current state rely on the results of previous states, which makes it impossible to calculate in parallel and the algorithm converges slowly. Oord *et al.* propose a faster and more efficient architecture WaveNet [Oord *et al.*, 2016a] for sequence learning. WaveNet combines causal filters with dilated convolutions to allow their receptive fields to grow exponentially with depth, which is significant to capture the long-range temporal dependencies in sequential data. Another work using convolutions to replace LSTMs for sequence modelling is proposed in [Gehring *et al.*, 2017]. The sequence-to-sequence architecture in this paper is based en-

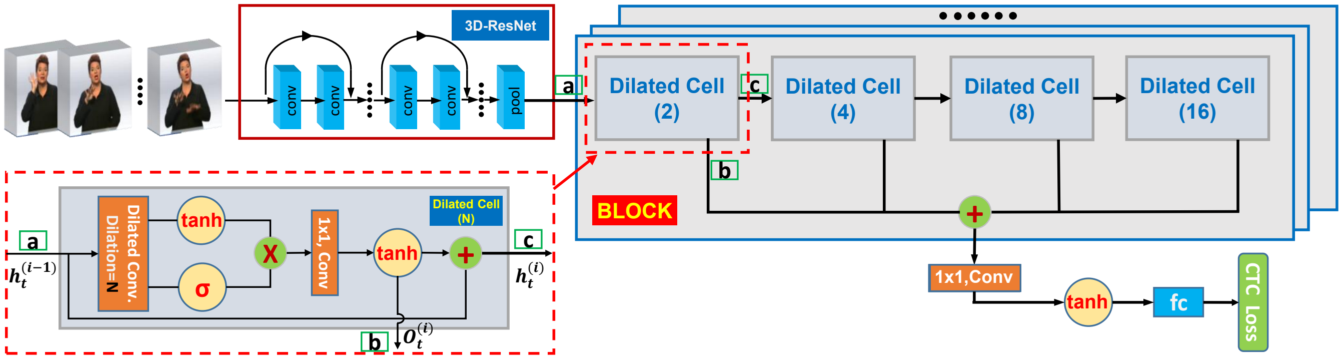


Figure 2: Overview of our sign language recognition framework. The system consists of a 3D residual network and a dilated convolutional network, followed by a CTC loss layer.

tirely on convolutional networks. Hence, the computations over all elements can be fully parallelized during training and the optimization is easier compared to the recurrent neural networks. This convolutional sequence-to-sequence model outperforms LSTMs for machine translation.

Recent works such as [Cui *et al.*, 2017; Camgoz *et al.*, 2017] employ a CNN-LSTM network with Connectionist Temporal Classification (CTC) [Graves *et al.*, 2006] for continuous sign language recognition. CTC approach has the capacity to train RNNs to label unsegmented sequences directly. Hence, it's appropriate to deal with continuous sign language translation due to the lack of supervision on accurate temporal segmentation for sign words. Cui *et al.* use a staged optimization strategy to train an end-to-end sequence learning scheme with the objective function of connectionist temporal classification [Cui *et al.*, 2017]. They apply the proposed method to a real-word continuous sign language recognition benchmark and it achieves the state-of-the-art performance. There are some other SLR approaches which combine the deep neural architecture and traditional sequential model [Koller *et al.*, 2016b; 2017]. Koller *et al.* embed an HMM into a deep recurrent CNN-BLSTM network [Koller *et al.*, 2017] with an iterative re-alignment approach for continuous sign language recognition. The hybrid CNN-HMM method iteratively treats the CNN outputs as Bayesian posteriors for HMM training and makes use of the hidden states of each frame predicted by HMM for CNN finetuning.

3 Our Method

In this section we propose a novel deep learning architecture for continuous sign language recognition (CSLR) with iterative optimization. The continuous sign language recognition system usually takes the video or image sequences as input, and automatically translates the visual sequences into natural language for easy understanding. Our CSLR architecture consists of two parts:

1. Visual Encoder for Feature Extraction: A residual 3D convolutional neural network (3D-ResNet) for video clip representation.
2. Sequence Learning Model: A dilated convolutional neural network with Connectionist Temporal Classification for sequence alignment and decoding.

We use Connectionist Temporal Classification (CTC) approach to train the Sequence Learning Network with sentence-level labels. After the convergence of the network learning, we get the alignment proposals between the video clips and sentence labels. With pseudo-labelled video clips, we fine-tune the visual feature extractor to get better representations of the visual inputs. Then we iteratively train and fine-tune these two networks. Figure 1 illustrates the iterative optimization strategy for our continuous sign language recognition system.

3.1 Network Architecture

The architecture of our proposed method is shown in Figure 2. Feeding in video clip sequence, our system consists of a 3D residual network for feature extraction, followed by a dilated convolutional neural network and a connectionist temporal classification layer for sequence learning and labelling.

3D Residual Network (3D-ResNet)

The 3D convolutional neural network has shown strong capability for video representation and achieves state-of-the-art results in action recognition [Ji *et al.*, 2013; Tran *et al.*, 2015].

3D convolutions not only model the spatial information, but consider the sequential relationship by temporal connections across frames. Considering the huge successes of Residual Networks in different image recognition tasks, we use 3D-ResNet, which only replace the 2D convolutional filters with 3D convolutional filters, to generate video representation.

Let $\mathbf{X} = (x_1, ..., x_T) = \{x_t\}_{t=1}^T$ denote an input sequence of images with T frames, a sliding window is performed on $\mathbf{X}$ to generate a video sequence $\mathbf{V}^N = (v_1, ..., v_N)$ of N clips. We use $\mathbf{\Phi}_\Theta(\cdot)$ to represent 3D-ResNet, where Θ are the network weights. Passing each video clip v_t through the 3D-ResNet to produce a fixed-length vector representation $f_t \in \mathbb{R}^d$, the input sequence is represented as

$$\mathbf{F}^N = (f_1, ..., f_N) = \{\mathbf{\Phi}_\Theta(v_t)\}_{t=1}^N. \quad (1)$$

In our experiments, we use the 18-layer 3D ResNet considering the low memory consumption and less computational cost. The architecture and convolutional filter size of 3D ResNet is shown in Figure 3. The activation of pooling layer with a dimension of 512 is extracted as feature representation.

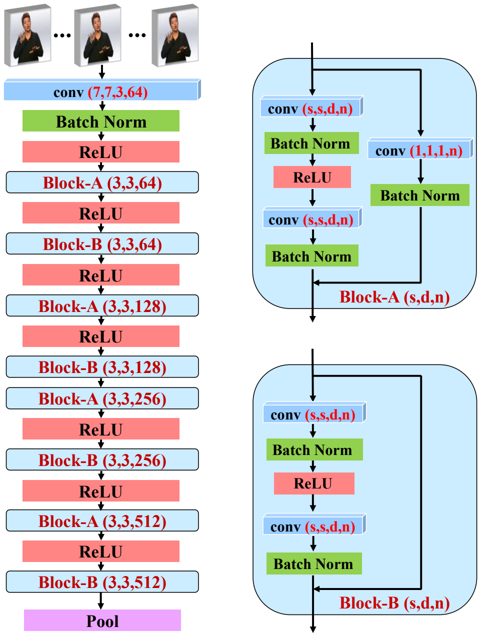


Figure 3: A 18-layer 3D residual convolutional network with two different skip connections.

Dilated Convolutions

Dilated convolutions have made huge success for audio generation in WaveNet [Oord *et al.*, 2016a]. When applying the key idea of WaveNet to continuous sign language recognition, it yields state-of-the-art performance. The filter of the dilated convolution is applied over an area larger than its length by skipping input values with a certain step.

Dilated convolutions with different dilations have different receptive fields. Stacked dilated convolutions enable network to have very large receptive fields with just a small number of layers, since the dilation range increases exponentially. This makes the network capture the temporal dependency with various resolutions for the input sequences. For each dilated convolutional layer, we employ the same gated activation unit as mentioned in gated PixelCNN [Oord *et al.*, 2016b] and WaveNet [Oord *et al.*, 2016a]. The outputs o_i and h_i of the i^{th} dilated convolutional cell for the t^{th} clip are

$$z = tanh(\mathcal{C}_d(h_t^{(i-1)})) \odot \sigma(\mathcal{C}_d(h_t^{(i-1)})), \tag{2}$$

$$o_t^{(i)} = tanh(\mathcal{C}_{1*1}(z)), \tag{3}$$

$$h_t^{(i)} = h_t^{(i-1)} + o_t^{(i)}, \tag{4}$$

where $\mathcal{C}_d$ and $\mathcal{C}_{1*1}$ stand for dilated convolution and 1×1 convolution, respectively. Denoting the stacked dilated convolutional cells as "BLOCK", the outputs of the dilated convolutional network are the sums of $o_t^{(i)}$ for all dilated cells and blocks, written as

$$o_t = \sum_{all-blocks} \sum_i o_t^{(i)}. \tag{5}$$

After all dilated BLOCKs, a 1x1 convolutional layer with the activation of $tanh$ is applied. At the end, we use a fully-connected layer to embed the outputs into non-normalized categorical probabilities of word-level labels with K classes:

$$\boldsymbol{y}_t = W_{fc} \times tanh(\mathcal{C}_{1*1}(o_t)) + b_{fc}. \tag{6}$$

The final probability distribution of a sequence with N clips can be written as

$$\boldsymbol{Y} = (Y_{ij}) = [\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_N]^T, \tag{7}$$

where Y_{ij} is the log-probability of label j at i^{th} clip.

3.2 Connectionist Temporal Classification (CTC)

Connectionist Temporal Classification (CTC) [Graves *et al.*, 2006] introduces a "blank" label ($-$), which means the input clip does not belong to any category in the vocabulary. Denote the intermediate label path of the input sequence as $\pi = (\pi_1, ..., \pi_T)$, where $\pi_t \in \mathcal{V} \bigcup \{-\}$, $\mathcal{V}$ is the sign word vocabulary. Given input $\mathbf{X}$, the probabilities $p(\pi|\mathbf{X})$ of π is

$$p(\pi|\mathbf{X}) = \prod_{t=1}^{T} \mathrm{P}(\pi_t|\mathbf{X}) = \prod_{t=1}^{T} Y_{t,\pi_t}. \tag{8}$$

Define a many-to-one map $\mathcal{B}$ which simply removes all blanks and repeated labels from the paths (e.g. $\mathcal{B}(-bb-eel-l-) = \mathcal{B}(b-el-l) = bell$). Thus, given the sentence-level sequence label $\boldsymbol{s} = (s_1, ..., s_L)$, where L is the the number of symbols in the sequence, the conditional probabilities of $\boldsymbol{s}$ is calculated by summing up the probabilities of all corresponding paths:

$$p(\boldsymbol{s}|\mathbf{X}) = \sum_{\pi \in \mathcal{B}^{-1}(\boldsymbol{s})} p(\pi|\mathbf{X}), \tag{9}$$

where $\mathcal{B}^{-1}(s) = \{\pi|\mathcal{B}(\pi) = \boldsymbol{s}\}$ is the inverse mapping function of $\mathcal{B}$.

The CTC loss is defined with the negative log-likelihood of the ground truth as

$$\mathcal{L}_{\text{CTC}} = -\ln p(\boldsymbol{s}|\mathbf{X}). \tag{10}$$

To efficiently compute the probability $p(\boldsymbol{s}|\mathbf{X})$, the forward-backward algorithm is applied. More details about forward-backward algorithm and the optimization procedure are introduced in [Graves *et al.*, 2006].

In order to obtain a better representation of 3D convolutional network of input clips, we fine-tune the 3D-ResNet with the labels generated from the dilated convolutional network and CTC layer after convergence. The label ℓ_i of the i^{th} clip is generated by

$$\ell_i = \arg\max_j Y_{i*}. \tag{11}$$

Such labels are called as pseudo-labels, since they are automatically generated by the model. When training the network, the CTC loss has limited contribution to CNN parameters due to the chain rules for back-propagation. Fine-tuning the ResNet with pseudo-labels will alleviate this issue, and obtain better representations of input clips.

4 Experiments

In this section we provide some experimental illustrations and extensive evaluations of our method on continuous sign language recognition dataset.

	Train	Test	Dev
#Sentences	5672	629	540
#Vocabulary	1231	497	461
#Words	65227	6530	5564

Table 1: Summary of RWTH-PHOENIX-Weather-2014 dataset.

4.1 Dataset and Evaluation

We conduct our experiments on RWTH-PHOENIX-Weather-2014 [Koller *et al.*, 2015], which is a popular benchmark dataset for continuous SLR in German Sign Language. This dataset provides RGB videos for full frames and cropped hand patches. The videos are performed by 9 signers with around 1 million frames and 6841 sentences in total. The statistic details of this dataset are available in Table 1.

We measure our system performance with Word Error Rate (WER), which is wildly used in speech recognition and machine translation system. The WER is some kind of performance metric derived from the Levenshtein distance. It measures the least operations of substitutions, deletion and insertion to transform the generated sequences into the reference sequence:

$$\text{WER} = \frac{\#\text{insertions} + \#\text{deletions} + \#\text{substitutions}}{\text{length of reference}}. \tag{12}$$

Based on the definition of Eq. 12, a lower WER means a better performance.

4.2 Iterative Optimization

We use iterative optimization strategy shown in Figure 1 to train our network. The step-by-step training procedure in detail is introduced in this section.

3D-ResNet Setups and Initialization

In our experiments, the input images are resized to 224×224. The 3D convolutional network requires fixed-length video clip inputs, so a sliding window is performed to generate clips from the raw input videos. The sliding window size is set to be 8, which is able to cover the sign word if the gloss exactly exists in the clip. In addition, we set the stride of the sliding window as 4, which means there are 50% overlaps between the adjacent clips. In this way, there are 190,536 clips in training set, 21,349 clips in testing set, and 17,908 clips in dev set. Each video includes around 34 clips in average.

As mentioned in the previous sections, the 3D-ResNet is trained by the supervision of pseudo-labels obtained from the sequence alignment procedure. In the initial step, we pretrain our 3D-ResNet on an isolated SLR dataset introduced in [Zhang *et al.*, 2016], using stochastic gradient descent optimizer (SGD) with a batch size of 5, a learning rate of 0.001, a momentum of 0.9, and a weight decay of 5×10^{-5}. After the convergence of the network with 155k iterations, the 512-D pooling activations are extracted as the clip representations.

Sequence Learning and Pseudo-label Generation

The dilated convolutional network is trained with CTC loss. Every block has the dilation of 1, 2, 4, 8, 16 for each layer, and the size of block is 5. The fully-connected layer before the softmax-layer has a size of 1024. We train the network using Adam optimizer [Kingma and Ba, 2014] with a learning rate of 1×10^{-4}. The batch size is 20, and the network coverages very fast within around 10 epochs.

For inference, we pass the test clip sequence through network and obtain the posterior probability distributions for each clip. The CTC beam search decoder in TensorFlow [Abadi *et al.*, 2016] is used to generate new sentence corresponding to the input video. We can get the softmax probability distribution for each training clip, and assign the most-likely sign word label which has the maximum probability to generate pseudo-labels. We alternately optimize the 3D-ResNet feature extractor and dilated convolutional network.

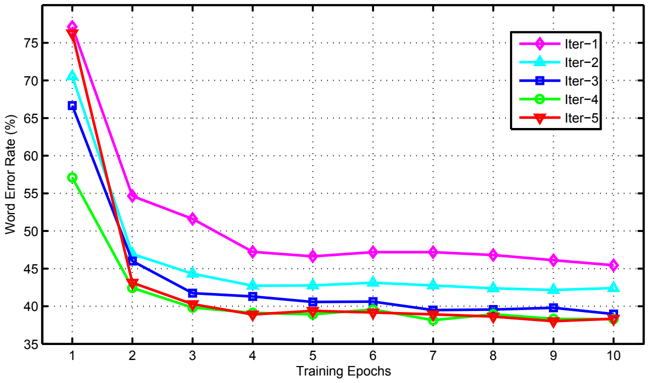


Figure 4: WER on RWTH-PHOENIX-Weather-2014 Dev set.

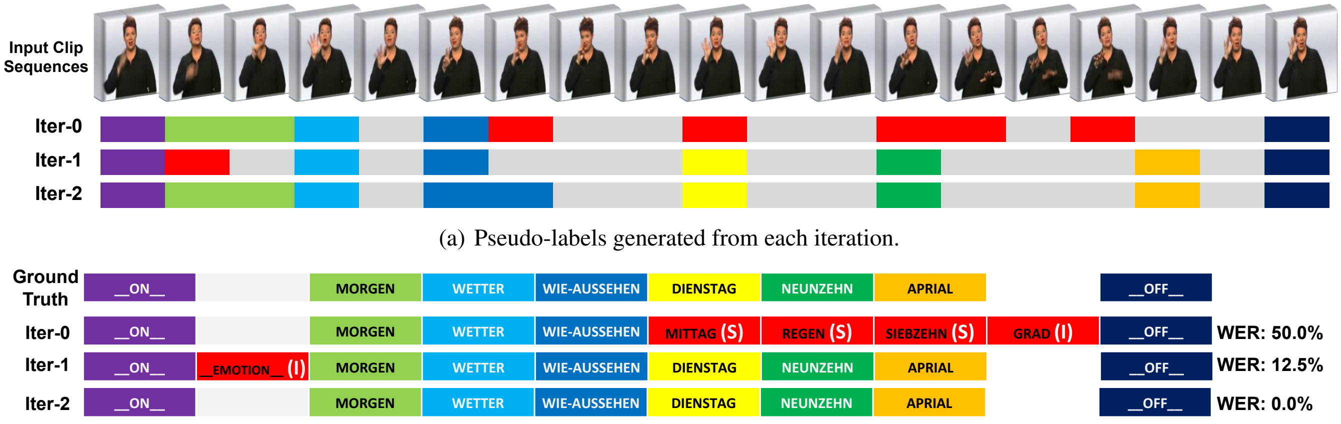
Iterations	Dev		Test	
	del / ins	WER	del / ins	WER
Iter-0	18.5 / 2.6	60.3	18.1 / 2.8	59.7
Iter-1	11.0 / 3.9	45.4	11.7 / 3.5	45.5
Iter-2	9.7 / 4.3	41.2	9.1 / 4.5	41.5
Iter-3	9.7 / 4.0	39.0	8.8 / 4.1	39.0
Iter-4	8.7 / 4.5	38.3	7.8 / 4.4	37.7
Iter-5	8.3 / 4.8	38.0	7.6 / 4.8	37.3

Table 2: Word error rate (WER) on RWTH-PHOENIX-Weather-2014 for different iterations (the lower the better).

4.3 Results

Iterative Results

Figure 4 shows how the word error rates on RWTH-PHOENIX-Weather-2014-Multisigner Dev set decrease with different epochs and iterations. From the results, we find that the system becomes convergence from the fourth epoch for each iteration. The network is fully trained since the performance curves of iteration-4 and iteration-5 getting much closer to each other. Hence, we stop training after the fifth iteration. The WERs on Test set and Dev set of RWTH-PHOENIX-Weather-2014-Multisigners are shown in Table 2. In this table, “ins” and “del” mean the average operations of “insertion” and “deletion” that transform the generated sentences into the sentences of ground-truth. The performances are getting better (lower WERs) with more training iterations.



(a) Pseudo-labels generated from each iteration.

(b) CTC beam search decoded results for each iterations.

Figure 5: An example for iterative optimization from iter-0 to iter-2. (a) The pseudo-labels for 3D-ResNet finetuning generated from each iteration. Red symbols mean that these words are not in the ground truth sentence. The wrongly recognized words are corrected step by step. (b) CTC beam search decoder results. "S" and "I" stand for the operations of "substitution" and "insertion", respectively that turn the recognised sentence to the reference sentence.

Methods	Dev		Test	
	del / ins	WER	del / ins	WER
1-Mio-Hands	16.3 / 4.6	47.1	15.2 / 4.6	45.1
SubUNet	14.6 / 4.0	40.8	14.3 / 4.0	40.7
CNN-Hybrid	12.6 / 5.1	38.3	11.1 / 5.7	38.8
Staged-Opt	13.7 / 7.3	39.4	12.2 / 7.5	38.7
Ours	8.3 / 4.8	**38.0**	7.6 / 4.8	**37.3**

Table 3: Word error rate (WER) on RWTH-PHOENIX-Weather-2014 (the lower the better).

Figure 5 shows an example for our iterative optimization algorithm. The gray symbols stand for "blank" labels for CTC training and decoding. Other symbols in colors (exclude red) mean the decoded words belong to the ground truth sentence. In addition, the words in red are not in the the ground truth sentence which means they're wrongly recognized. Figure 5(a) illustrates the pseudo-labels generation from sequence learning stage The results of CTC beam search decoder for the same example are shown in Figure 5(b). At iteration-0, the decoded sentence is not absolutely right, with four wrongly recognized words, resulting in a low WER of 50%. After finetuning the 3D-ResNet, we re-train our sequence learning network again, the WER of the sentence decoded in iteration-1 gets much lower. Further, we get a completely right result after the second iteration.

We conduct experiments on a single Nvidia GTX 1080Ti GPU. Inference time depends on the video length. In average, it takes around 1 second to recognize a sign video with 140 frames (850ms for feature extraction and 150ms for CTC beam search decoder with a beam width of 100).

Comparison with the state-of-the-art

In this part, we evaluate the performance of our method by comparing it to some existing algorithms on RWTH-PHOENIX-Weather-2014-Multisigner dataset. The performance comparisons are summarized in Table 3. 1-Mio-Hands [Koller *et al.*, 2015; 2016a] embeds a CNN within an iterative EM algorithm. SubUNet [Camgoz *et al.*, 2017] and Staged-Optimization [Cui *et al.*, 2017] both use the BLSTM+CTC framework, and achieve WERs of 40.8% and 39.4% on dev set, 40.7% and 38.7% on test set, respectively. The hybrid CNN-HMM [Koller *et al.*, 2016b] combines the discriminative abilities of CNNs with the sequence modelling capabilities of HMM, and achieves the WERs of 38.3% and 38.8% on dev and test set. As seen from the results, our method outperforms the state-of-the-art by 1.4% on test set, with a lower WER of 37.3%. With iterative optimization, our proposed architecture achieves the state-of-the-art performance.

5 Conclusion

This paper presents a deep learning framework for continuous sign language recognition based on 3D-ResNet and dilated convolutional network, with an iterative optimization strategy. We use connectionist temporal classification approach to align each clip to its corresponding gloss label in sentence, and utilize these generated alignment proposals (so-called pseudo-labels) to fine-tune our feature extractor. We alternately optimize our 3D-ResNet for feature extraction and dilated convolutional network for pseudo-label generation. We conduct experiments on a large continuous sign language benchmark RWTH-PHOENIX-Weather dataset. Our approach outperforms the state-of-the-art with a lower WER, which demonstrates the effectiveness of our method.

Acknowledgments

This work was supported in part to Dr. Houqiang Li by 973 Program (No. 2015CB351803) and NSFC (No. 61390514), and in part to Dr. Wengang Zhou by NSFC under contract No. 61472378 and No. 61632019, the Fundamental Research Funds for the Central Universities, and Young Elite Scientists Sponsorship Program By CAST (2016QNRC001).

References

[Abadi *et al.*, 2016] Martín Abadi, Paul Barham, Jianmin Chen, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.

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

[Cho *et al.*, 2014] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[Cui *et al.*, 2017] Runpeng Cui, Hu Liu, and Changshui Zhang. Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. In *CVPR*, pages 7361–7369, 2017.

[Gehring *et al.*, 2017] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. *arXiv preprint arXiv:1705.03122*, 2017.

[Graves *et al.*, 2006] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*, pages 369–376, 2006.

[Guo *et al.*, 2018] Dan Guo, Wengang Zhou, Meng Wang, and Houqiang Li. Hierarchical lstm for sign language translation. In *AAAI*, 2018.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[Huang *et al.*, 2015] Jie Huang, Wengang Zhou, Houqiang Li, and Weiping Li. Sign language recognition using 3d convolutional neural networks. In *ICME*, pages 1–6, 2015.

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

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

[Kim *et al.*, 2017] Kyung-Min Kim, Min-Oh Heo, Seong-Ho Choi, and Byoung-Tak Zhang. Deepstory: video story qa by deep embedded memory networks. In *IJCAI*, 2017.

[Kingma and Ba, 2014] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2014.

[Koller *et al.*, 2015] 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.

[Koller *et al.*, 2016a] 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 *CVPR*, pages 3793–3802, 2016.

[Koller *et al.*, 2016b] Oscar Koller, O Zargaran, Hermann Ney, and Richard Bowden. Deep sign: Hybrid cnn-hmm for continuous sign language recognition. In *BMVC*, 2016.

[Koller *et al.*, 2017] Oscar Koller, Sepehr Zargaran, and Hermann Ney. Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms. In *CVPR*, 2017.

[Li *et al.*, 2017] Xuelong Li, Bin Zhao, and Xiaoqiang Lu. Mam-rnn: Multi-level attention model based rnn for video captioning. In *IJCAI*, 2017.

[Oord *et al.*, 2016a] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, et al. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[Oord *et al.*, 2016b] Aaron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray Kavukcuoglu. Conditional image generation with pixelcnn decoders. In *NIPS*, 2016.

[Pigou *et al.*, 2015] Lionel Pigou, Aäron van den Oord, Sander Dieleman, Mieke Van Herreweghe, and Joni Dambre. Beyond temporal pooling: Recurrence and temporal convolutions for gesture recognition in video. *IJCV*, pages 1–10, 2015.

[Pu *et al.*, 2016] Junfu Pu, Wengang Zhou, and Houqiang Li. Sign language recognition with multi-modal features. In *PCM*, pages 252–261, 2016.

[Qiu *et al.*, 2017] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *ICCV*, volume 1, page 8, 2017.

[Song *et al.*, 2017] Jingkuan Song, Zhao Guo, Lianli Gao, Wu Liu, Dongxiang Zhang, and Heng Tao Shen. Hierarchical lstm with adjusted temporal attention for video captioning. In *IJCAI*, 2017.

[Starner *et al.*, 1998] Thad Starner, Joshua Weaver, and Alex Pentland. Real-time american sign language recognition using desk and wearable computer based video. *TPAMI*, 20(12):1371–1375, 1998.

[Tran *et al.*, 2015] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015.

[Wang *et al.*, 2006] Sy Bor Wang, Ariadna Quattoni, L-P Morency, David Demirdjian, and Trevor Darrell. Hidden conditional random fields for gesture recognition. In *CVPR*, volume 2, pages 1521–1527, 2006.

[Zhang *et al.*, 2016] Jihai Zhang, Wengang Zhou, Chao Xie, Junfu Pu, and Houqiang Li. Chinese sign language recognition with adaptive hmm. In *ICME*, pages 1–6, 2016.

[Zhao *et al.*, 2017] Zhou Zhao, Qifan Yang, Deng Cai, Xiaofei He, and Yueting Zhuang. Video question answering via hierarchical spatio-temporal attention networks. In *IJCAI*, volume 2, 2017.