Brief Introduction to Continuous Sign Language Recognition

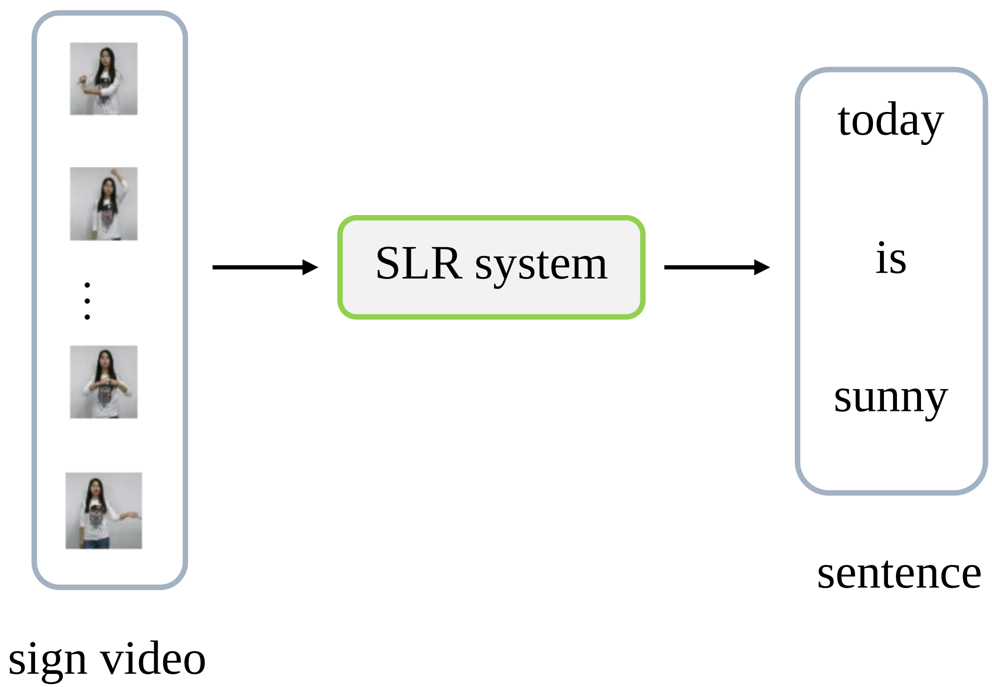
魏承承

2019. 1. 19

Introduction

- What does a continuous **sign language recognition (SLR)** system do?

word vocabulary: apple, sun, today, catch, you ……



Introduction

- Evaluation on Continuous SLR
  - Word Error Rate (WER)

$$\mathrm{WER} = \frac{\#\mathrm{sub} + \#\mathrm{del} + \#\mathrm{ins}}{\#\mathrm{words\ in\ reference}},$$

For example,

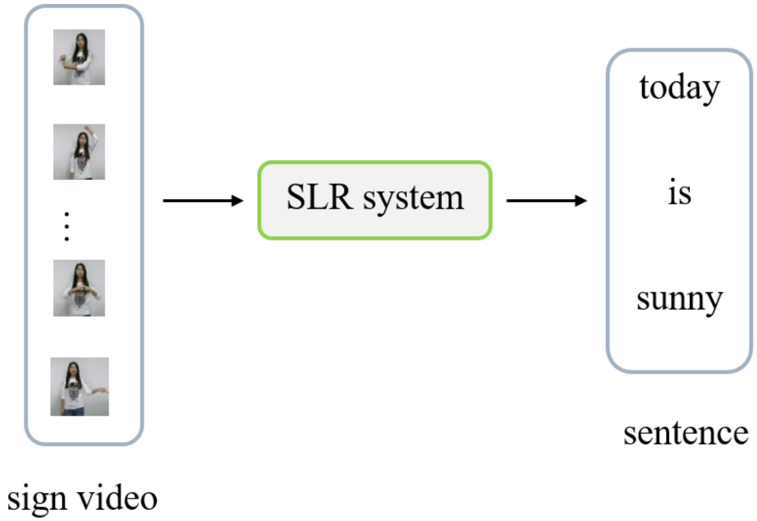
prediction: I (have) a cat ~~that~~ named Jerry.

groundtruth: I have a cat named Tom.

Calculate the WER: $\frac{1+1+1}{6}=0.5$

Introduction

- Continuous SLR is weakly-supervised
- 解决 Continuous SLR 问题的主流思路
  - 受语音识别领域启发：对每一帧识别，合并结果
    - ✓ Connectionist Temporal Classification（CTC）
    - ✓ CNN-RNN-CTC framework
  - 受机器翻译领域启发：从特征序列映射到文本序列
    - ✓ Encoder-Decoder framework



Introduction

- CTC: 逐一识别，再合并

Graves A, Fernández S, Gomez F, et al. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. ICML 2006

Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization [CVPR 2017]

Runpeng Cui* Hu Liu* Changshui Zhang
Department of Automation, Tsinghua University
State Key Laboratory of Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology (TNList), Beijing, China
{crp16@mails, liuhu15@mails, zcs@mail}.tsinghua.edu.cn

Framework : Spatio-temporal CNN - BLSTM - CTC

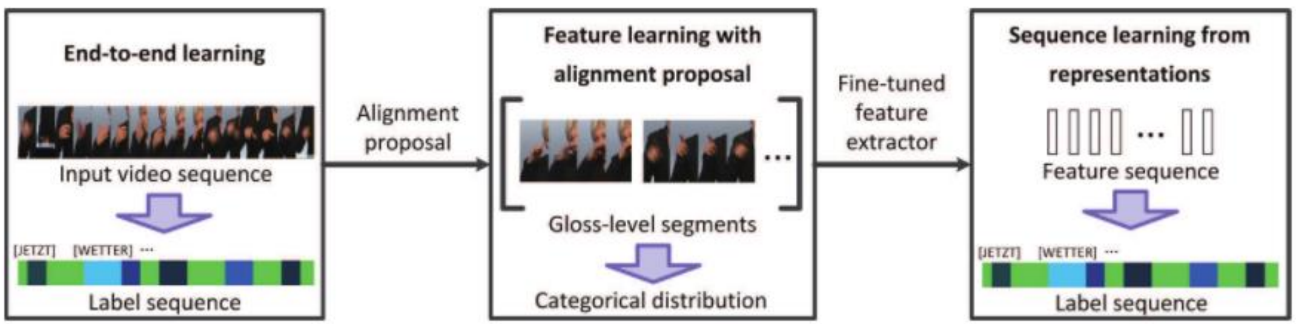


Figure 1. This is the overview of our staged training approach: (1) end-to-end-training the full architecture with feature and sequence learning components to predict the alignment proposal; (2) training the feature extractor with the alignment proposal; (3) training sequence learning component with the improved representation sequence as input, which is given by the fine-tuned feature extractor.

Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization [CVPR 2017]

- Step1: end-to-end learning

Spatio-temporal feature extractor
CNN (VGG-S / GoogLeNet)
conv1D-3-1024 maxpool1D-2 conv1D-3-1024 maxpool1D-2

d×N

Recurrent neural net
BLSTM-512 fully connected layer softmax

(K+1)×N

CTC

Conv1D: 沿时间维度卷积

$$\Pr(\pi|\boldsymbol{x}) = \prod_{n=1}^{N} \Pr(\pi_n|\boldsymbol{x}) = \prod_{n=1}^{N} P^{\mathrm{c}}_{\pi_n,n}, \tag{6}$$

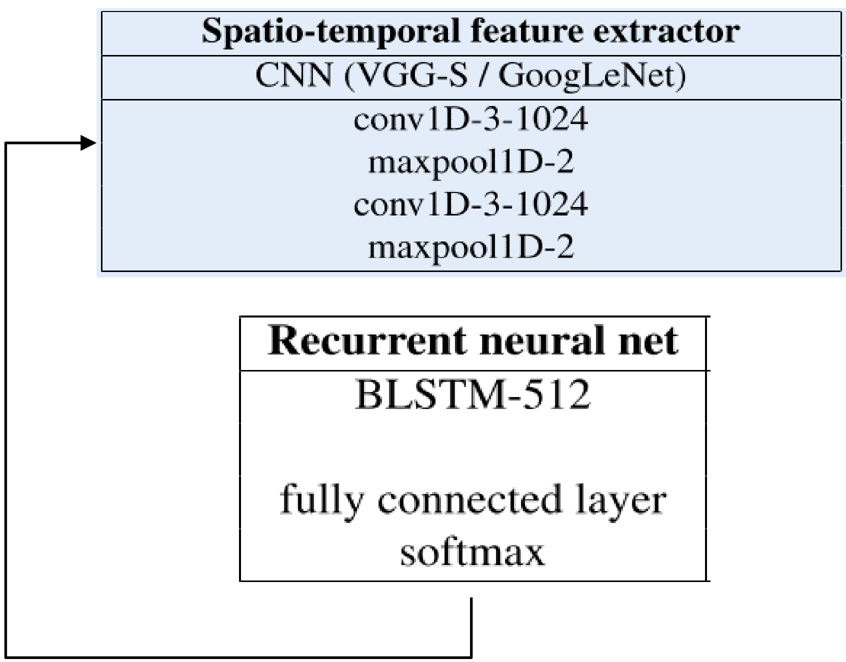
$$\Pr(\boldsymbol{y}|\boldsymbol{x}) = \sum_{\pi \in \mathcal{B}^{-1}(\boldsymbol{y})} \Pr(\pi|\boldsymbol{x}), \tag{7}$$

$$\mathcal{L}_{\mathrm{CTC}}(\boldsymbol{x}, \boldsymbol{y}) = -\log \Pr(\boldsymbol{y}|\boldsymbol{x}). \tag{8}$$

$$\mathcal{L} = \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{|\mathcal{S}|} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{S}} \mathcal{L}_{\mathrm{CTC}}(\boldsymbol{x}, \boldsymbol{y}), \tag{9}$$

Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization [CVPR 2017]

- Step2: Feature learning with alignment proposal
  - alignment proposal: output of BLSTM
  - to finetune the spatio-temporal feature extractor



$$\mathcal{L}_{\text{align}}(\boldsymbol{x}, P^{\alpha}(\boldsymbol{x})) = \frac{1}{N} \sum_{n=1}^{N} d_{\text{KL}}(p_n \| \varphi(s_n)), \quad (11)$$

$$\mathcal{L} = \frac{1}{|\mathcal{S}|} \sum_{(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{S}} \mathcal{L}_{\text{align}}(\boldsymbol{x}, P^{\alpha}(\boldsymbol{x})). \quad (12)$$

Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization [CVPR 2017]

Step3: Sequence learning from representations

Spatio-temporal feature extractor
CNN (VGG-S / GoogLeNet)
conv1D-3-1024 maxpool1D-2 conv1D-3-1024 maxpool1D-2

Recurrent neural net	**Detection net**
BLSTM-512	conv1D-2-256
	conv1D-2-256
fully connected layer	fully connected layer
softmax	softmax

CTC

$$z_k = \sum_{n=1}^{N} P_{kn}^{\mathrm{c}} \cdot P_{kn}^{\mathrm{d}}, \tag{13}$$

$$\mathcal{L}_{\mathrm{det}}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{k \in \mathcal{A} \setminus \mathcal{Y}} \log(1 - z_k) + \sum_{k \in \mathcal{Y}} \log z_k. \tag{14}$$

$$\mathcal{L} = \frac{\lambda}{2} \|\boldsymbol{w}\|^2 + \frac{1}{|\mathcal{S}|} \sum_{(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{S}} (\mathcal{L}_{\mathrm{CTC}} + \mu \mathcal{L}_{\mathrm{det}})(\boldsymbol{x}, \boldsymbol{y}), \tag{15}$$

Experimental results

Model setup	Validation del / ins / WER	Test del / ins / WER
Our-end2end	16.3 / 6.7 / 46.2	15.1 / 7.4 / 46.9
RNN	19.6 / 5.4 / 45.0	18.1 / 6.2 / 44.8
LSTM	18.1 / 5.7 / 43.3	17.1 / 6.6 / 43.6
BLSTM	14.9 / 6.7 / 41.4	15.1 / 7.1 / 41.9
BLSTM+det net	13.7 / 7.3 / **39.4**	12.2 / 7.5 / **38.7**

Table 3. Recognition results for sequence learning stage on RWTH-PHOENIX-Weather 2014 multi-signer dataset in [%]. We assess the performance of different recurrent models and our proposed detection net. "BLSTM+det net" stands for the employed model with bidirectional LSTM and detection net, and "Our-end2end" for the full model with best performance in the stage of end-to-end training.

Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization [CVPR 2017]

Comparisons

Model setup	Extra supervision	Modality			Validation		Test	
		r-hand	traj	face	del / ins	WER	del / ins	WER
HOG-3D [16]		✓			25.8 / 4.2	60.9	23.2 / 4.1	58.1
[16] CMLLR		✓	✓	✓	21.8 / 3.9	55.0	20.3 / 4.5	53.0
1-Mio-Hands [18]	✓	✓			19.1 / 4.1	51.6	17.5 / 4.5	50.2
1-Mio-Hands [18]+[16]	✓	✓	✓	✓	16.3 / 4.6	47.1	15.2 / 4.6	45.1
CNN-Hybrid [19]	✓	✓			12.6 / 5.1	**38.3**	11.1 / 5.7	38.8
Our-end2end		✓			16.3 / 6.7	46.2	15.1 / 7.4	46.9
Ours		✓			13.7 / 7.3	39.4	12.2 / 7.5	**38.7**

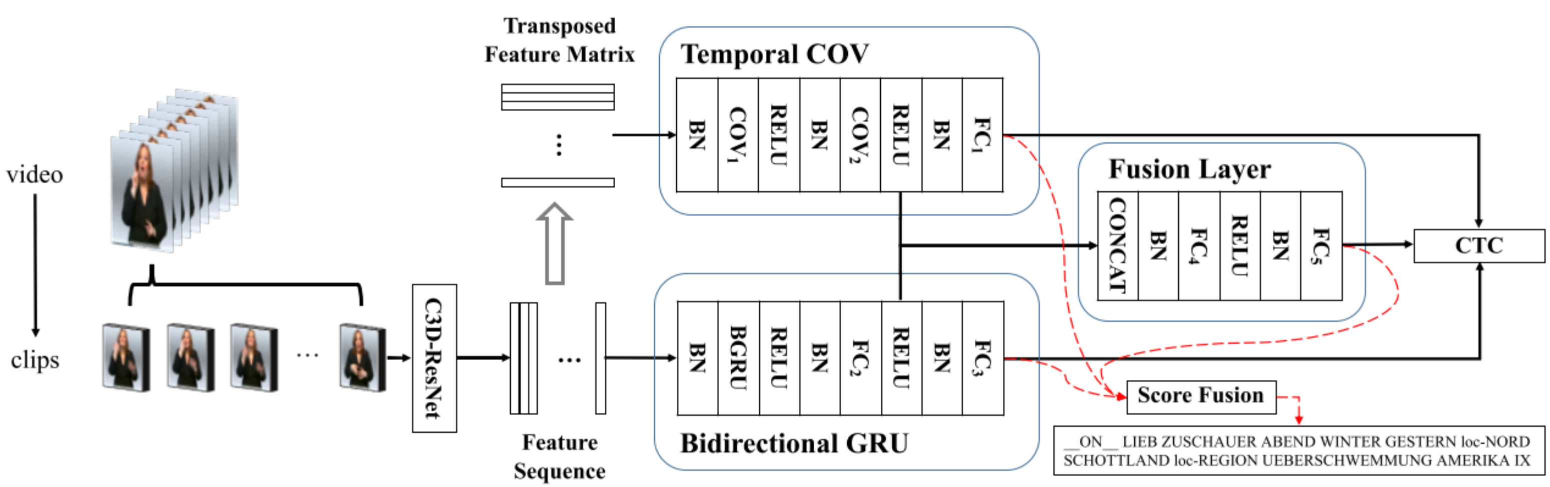
Table 4. Performance comparison of different continuous sign language recognition approaches on RWTH-PHOENIX-Weather 2014 multi-signer dataset in [%]. "r-hand" stands for right hand and "traj" stands for trajectory motion. "Extra supervision" imported in [18] contains a sign language lexicon mapping signs to hand shape sequences, and the best result of [19] uses [18]+[16] as the initial alignment.

Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization [CVPR 2017]

- Motivated by this paper…
  - alignment proposal: probability distribution -> argmax-> word
  - a staged optimization -> more staged optimization
  - ……

Connectionist Temporal Fusion for Sign Language Translation [MM2019]

Shuo Wang[1], Dan Guo[1*], Wen-Gang Zhou[2], Zheng-Jun Zha[2], Meng Wang[1]
[1] School of Computer and Information Engineering, Hefei University of Technology
[2] University of Science and Technology of China
shuowang.hfut@gmail.com,guodan@hfut.edu.cn,{zhwg,zhazj}@ustc.edu.cn,eric.mengwang@gmail.com



Connectionist Temporal Fusion for Sign Language Translation [MM2019]

- Temporal COV

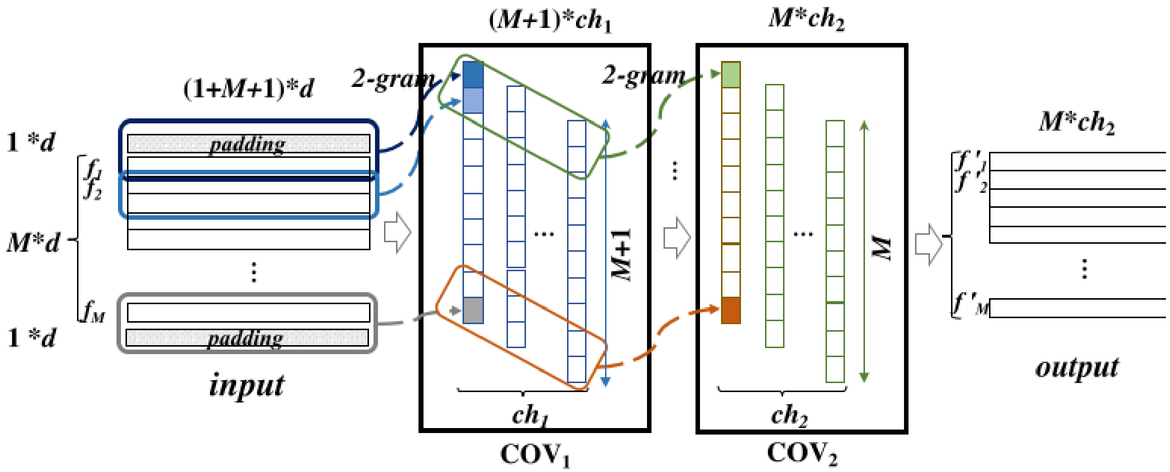


Figure 2: Illustration of temporal convolution operations in the TCOV module. With the filter number ch_1 and ch_2, we learn the feature embedding transformation with twice 2-gram (2-item) temporal convolution operations.

Connectionist Temporal Fusion for Sign Language Translation [MM2019]

Optimization

$$\mathcal{L} = \rho_1 \mathcal{L}_{CTC}(tcov) + \rho_2 \mathcal{L}_{CTC}(bgru) + \rho_3 \mathcal{L}_{CTC}(fl) \qquad (8)$$

Decoding

- argmax-> delete blank -> delete continuous repetitions

Connectionist Temporal Fusion for Sign Language Translation [MM2019]

- experimental result

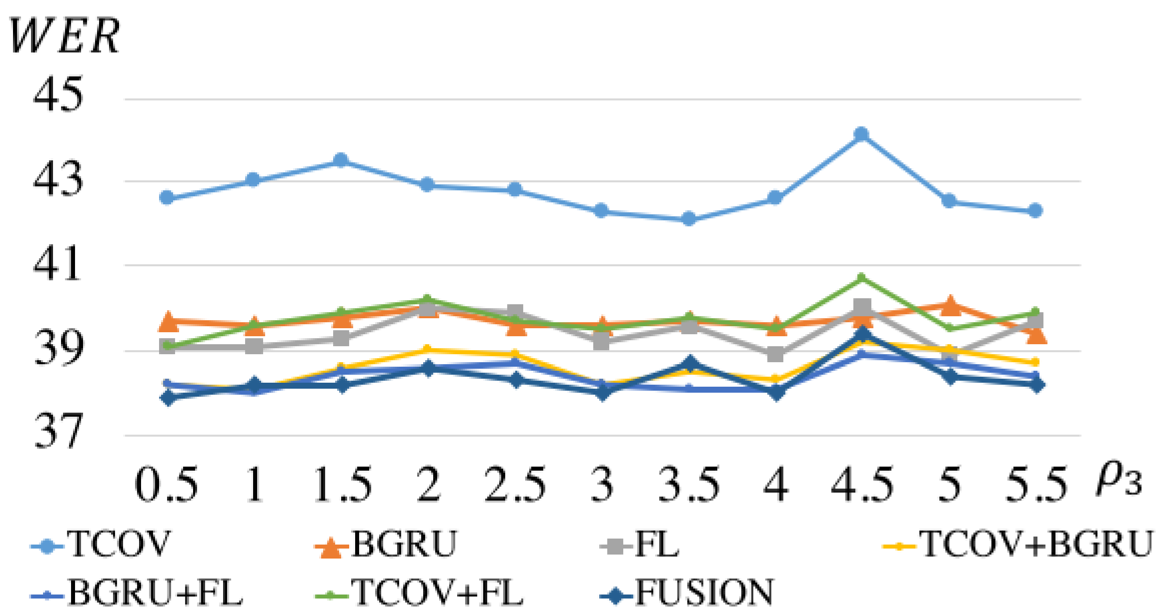
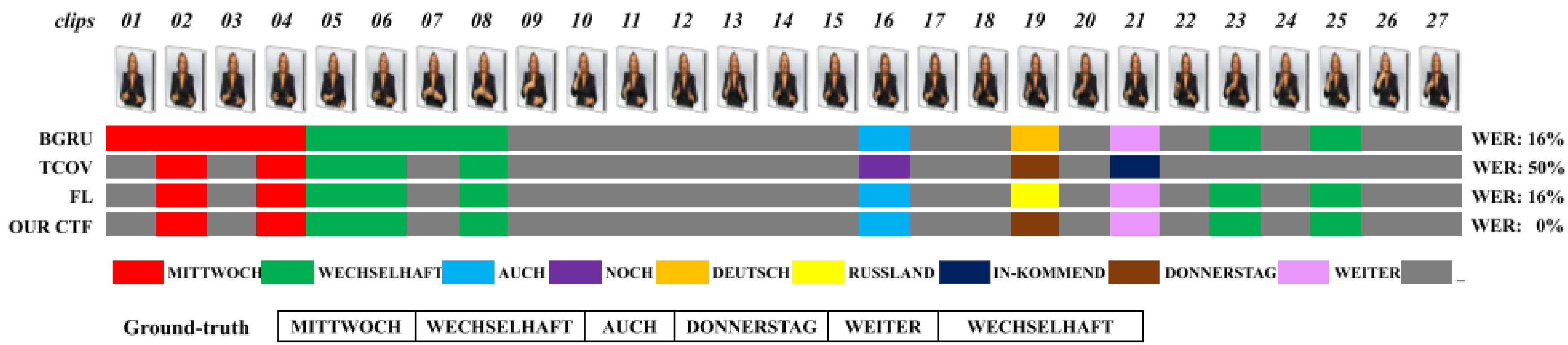


Figure 4: The Performances on hyper-pamamter ρ_3.

Connectionist Temporal Fusion for Sign Language Translation [MM2019]

experimental result



Fusion Module Set	VAL			TEST		
	del	ins	WER	del	ins	WER
{TCOV}	14.9	6.3	42.6	14.4	6.5	41.6
{BGRU}	10.8	7.3	39.7	9.8	8.1	39.9
{FL}	11.5	5.8	39.1	11.1	6.4	39.4
{TCOV, BGRU}	13.3	5.5	38.2	12.0	5.9	38.1
{(TCOV, FL}	13.5	5.4	39.1	12.9	5.4	38.9
{BGRU, FL}	11.6	5.8	38.2	10.7	6.5	38.5
{TCOV, BGRU, FL}	12.8	5.2	**37.9**	11.9	5.6	**37.8**

Connectionist Temporal Fusion for Sign Language Translation [MM2019]

Comparisons

Table 6: Compared with other existing methods on RWTH-PHOENIX-Weather Dataset.

Method	Extra supervision	Modality			VAL		TEST	
		r-hand	traj	face	del / ins	WER	del / ins	WER
HOG-3D [18]		√			25.8 / 4.2	60.9	23.2 / 4.1	58.1
CMLLR [18]		√	√	√	21.8 / 3.9	55.0	20.3 / 4.5	53.0
1-Mio-Hands [19]	√	√			19.1 / 4.1	51.6	17.5 / 4.5	50.2
1-Mio-Hands [18, 19]	√	√	√	√	16.3 / 4.6	47.1	15.2 / 4.6	45.1
CNN-Hybrid [20]	√	√			12.6 / 5.1	38.3	11.1 / 5.7	38.8
Staged Optimization [5]		√			13.7 / 7.3	39.4	12.2 / 7.5	38.7
SubuNets [2]		√			14.6 / 4.0	40.8	14.3 / 4.0	40.7
Dilated CNN [26]					8.3 / 4.8	38.0	7.6 / 4.8	**37.3**
LS-HAN [16]					-	-	-	38.3
OUR CTF					12.8 / 5.2	**37.9**	11.9 / 5.6	37.8

The end

Thank you