System Description of T266 Team System for CNVSRC Challenge 2023

Shengqiang Li, Chao Lei, Baozhong Ma, Binbin Zhang, Fuping Pan

General User Agent, Shanghai, China

{shengqiang.li, chao.lei, baozhong.ma, binbin.zhang, fuping.pan}@guasemi.com

Abstract

This study describes our system for Task 1 Single-speaker Visual Speech Recognition (VSR) *fixed track* in the Chinese Continuous Visual Speech Recognition Challenge (CNVSRC) 2023. Specifically, we use intermediate connectionist temporal classification (Inter CTC) residual modules to relax the conditional independence assumption of CTC in our model. Then we use a bi-transformer decoder to enable the model to capture both past and future contextual information. In addition, we use Chinese characters as the modeling units to improve the recognition accuracy of our model. Finally, we use a recurrent neural network language model (RNNLM) for shallow fusion in the inference stage. Experiments show that our system achieves a character error rate (CER) of 38.09% on the Eval set which reaches a relative CER reduction of 21.63% over the official baseline.

1. Data

In this section, we provide a complete description of the data profile used to model training. Specifically, for Task 1 Single-speaker VSR *fixed track*, only *CN-CVS* [1] and *CNVSRC-Single.Dev* are used to perform system development. The train set of *CN-CVS* contains about 252 hours of video (175,058 utterances). The validation set of *CN-CVS* contains about 1.4 hours of video (913 utterances). The test set of *CN-CVS* contains about 1.4 hours of video (903 utterances). The train set of *CNVSRC-Single.Dev* contains about 90.6 hours of video (25,038 utterances). The validation set of *CNVSRC-Single.Dev* contains about 2 hours of video (568 utterances).

2. System

Figure 1 shows the proposed VSR model, which consists of a visual front-end, 12 conformer encoder blocks with 3 Inter CTC residual modules, a bi-transformer decoder and a CTC layer. In this section, we will provide a complete description of the model structure along with their vital configurations.

2.1. Visual Front-end

The visual front-end network [2] transforms input video frames into temporal sequences. A 3D convolution stem with kernel size $5 \times 7 \times 7$ is first applied to the video. Each video frame is then processed independently using a 2D ResNet-18 [3] with an output spatial average pooling. Temporal features are then projected to the back-end network input dimension using a linear layer.

2.2. Back-end Networks

The back-end networks use a Conformer architecture [2, 4] as the building blocks. In addition, Inter CTC residual modules

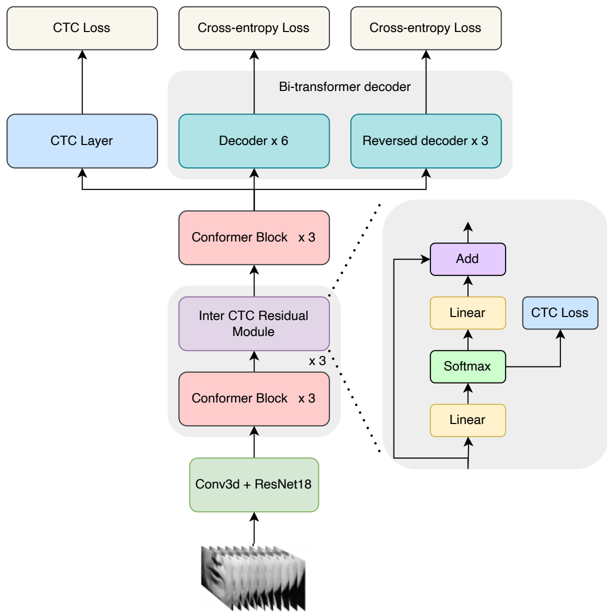

Figure 1: Architecture of the proposed VSR model. Intermediate predictions in Inter CTC residual module are summed to the input of the next Conformer block to condition the prediction of the final block on it. Intermediate CTC losses are added to the output CTC loss for the computation of the final loss.

and a Bi-transformer decoder are used to improve the performance of the model.

2.2.1. Intermediate CTC residual module

We add Inter CTC residual modules [5, 6, 7] in encoder networks. We condition intermediate block features of visual encoders on early predictions to relax the conditional independence assumption of CTC models. During both training and inference, each intermediate prediction is summed to the input of the next layer to help recognition. The output feature of the l-th encoder block $\boldsymbol{X}_{\text{out}}^{l}$ is passed through a feed-forward network with residual connection and a softmax activation function:

$$\begin{aligned} \boldsymbol{Z}_l &= \text{Softmax}\left(\text{Linear}\left(\boldsymbol{X}_{\text{out}}^{l}\right)\right) \\ \boldsymbol{X}_{\text{in}}^{l+1} &= \boldsymbol{X}_{\text{out}}^{l} + \text{Linear}\left(\boldsymbol{Z}_l\right) \end{aligned} \tag{1}$$

Where $\boldsymbol{X}_{\text{in}}^{l+1}$ is the input feature of the next encoder block, $\boldsymbol{Z}_l \in \mathbb{R}^{T \times V}$ is a probability distribution over the output vocabulary. The intermediate CTC loss of k-th Inter CTC residual module $\mathcal{L}_{\text{inter}}^{k}$ is then computed using the target sequence $\boldsymbol{y}$ as:

$$\begin{aligned} \mathcal{L}_{\text{inter}}^{k} &= -\log\left(p\left(\boldsymbol{y} \mid \boldsymbol{Z}^{l}\right)\right) \\ \text{with } p\left(\boldsymbol{y} \mid \boldsymbol{Z}^{l}\right) &= \sum_{\pi \in \mathcal{B}_{CTC}^{-1}(\boldsymbol{y})} \prod_{t=1}^{T} \boldsymbol{Z}_{t,\pi_t} \end{aligned} \tag{2}$$

Where $\pi \in V^T$ are paths of tokens and $\mathcal{B}_{CTC}$ is a many-to-one map that simply removes all blanks and repeated labels. The total intermediate CTC loss $\mathcal{L}_{\text{inter}}$ is computed as:

$$\mathcal{L}_{\text{inter}} = \frac{1}{K}\sum_{k=1}^{K} \mathcal{L}_{\text{inter}}^{k} \tag{3}$$

Where K is the total number of the Inter CTC residual modules in the encoder. We use Inter CTC residual module for every three conformer blocks with K set to 3.

2.2.2. *Bi-transformer Decoder*

Inspired by our early work [8, 9] on ASR task, we use the bi-transformer decoder to enable the model to capture both past and future contextual information. During inference, we use the left decoder only. During training, the loss function of the left decoder $\mathcal{L}_{left}$ is computed as:

$$\mathcal{L}_{left} = -\log\left(\prod_{l=1}^{L} p\left(y_l \mid \boldsymbol{y}_{1:l-1}, \boldsymbol{X}_{\text{e}}\right)\right) \tag{4}$$

The loss function of the right decoder $\mathcal{L}_right$ is computed as:

$$\mathcal{L}_{right} = -\log\left(\prod_{l=L}^{1} p\left(y_l \mid \boldsymbol{y}_{L:l-1}, \boldsymbol{X}_{\text{e}}\right)\right) \tag{5}$$

Where $\boldsymbol{y} = (y_1, \cdots, y_L)$ denotes the target sequence, and $\boldsymbol{X}_{\text{e}}$ denotes the output of the encoder. The total loss function of the bi-transformer decoder is:

$$\mathcal{L}_{attn} = (1-\alpha)\,\mathcal{L}_{left} + \alpha\mathcal{L}_{right} \tag{6}$$

Where α is a tunable hyper-parameter. In our system, α is set to 0.3.

2.3. Objective Function

The loss function of the proposed vsr model is computed as:

$$\mathcal{L} = \lambda\left(\gamma\mathcal{L}_{inter} + (1-\gamma)\,\mathcal{L}_{ctc}\right) + (1-\lambda)\,\mathcal{L}_{attn} \tag{7}$$

where $\mathcal{L}_{ctc}$ is an auxiliary CTC loss function [10], and λ and γ are two tunable hyper-parameters. In our system, λ is set to 0.1 and γ is set to 0.3.

3. Experiments

3.1. Pre-processing

In this section, we provide a complete description of the data processing pipeline. The RetinaFace [11] face detector and Face Alignment Network (FAN) [12] are used to detect 68 facial landmarks. Similar to [2], we remove differences related to rotation and scale by cropping the lip regions using bounding boxes of 96 × 96 pixels to facilitate recognition. The cropped images are then converted to gray-scale and normalised between -1 and 1. In the training stage, video streams are augmented with random cropping and adaptive time masking [13]. We use Chinese characters as modeling units, the vocabulary consists of 4466 Chinese characters generated from the text of the train set of *CN-CVS* and CNVSRC-Single.Dev, and 3 special tokens for $< blank >, < unk >, < eos/sos >$.

3.2. Experimental settings

All of our experiments were implemented using CNVSRC 2023 Baseline[1]. Following the default set in CNVSRC 2023 Baseline, we use 12 conformer layers in the encoder where the attention dimension is 768, the number of attention heads is 12, the kernel size of the cnn module is 31, the feed-forward network dimension is 3072. The bi-transformer decoder consists of 6 transformer decoders and 3 reversed transformer decoders, where the attention dimension is 768, the number of attention heads is 12 and the feed-forward network dimension is 3072. The RNN langugae model has two layers and the hidden size of each layer is 650.

3.3. Training

The vsr model is trained from scratch through curriculum learning. Firstly, we train the vsr model using the subset that includes only short utterances lasting no more than 4 seconds of *CN-CVS* and average the model weights from epoch 14 to epoch 23. Secondly, we use the averaged checkpoint from stage 1 to initialize the vsr model and train the vsr model with the full dataset of *CN-CVS* and average the model weights from epoch 65 to epoch 74. Finally, we use the averaged checkpoint from stage 2 to initialize the vsr model and train the vsr model with the train set of *CNVSRC-Single.Dev*. For the main results shown in Table 1, the model is trained using 25038 utterances of *CNVSRC-Single.Dev* as the train set and 568 utterances of *CNVSRC-Single.Dev* as the valid set, which is the same as mentioned in Section 1. For the ablation study shown in Table 2, the model is trained using 22767 utterances of *CNVSRC-Single.Dev* as the train set and 2839 utterances of *CNVSRC-Single.Dev* as the valid set, the same as the CNVSRC 2023 Baseline. The RNN language model is trained using the train set of *CN-CVS* and *CNVSRC-Single.Dev* for 60 epochs.

3.4. Inference

Decoding is performed using joint CTC/attention one-pass decoding [14], and an RNN language model is used for shallow fusion. During decoding, the CTC weight is 0.3, the lm weight is 0.1 and the beam size 40.

3.5. Results

3.5.1. *Main Results*

Table 1 reports the character error rate (CER) of the proposed system on the eval set. From the table, we see that our system achieves a CER of 38.09%, which reaches a relative CER reduction of 21.63% over the official baseline.

3.5.2. *Ablation study*

Table 2 shows the ablation study of the proposed system on the valid set. Valid set 1 contains 2839 utterances, is the same as CVSRC2023 Baseline. Valid set 2 contains 568 utterances, is a subset of the valid set 1. D1 denotes that the train set and the valid set are the same as the CVSRC2023 Baseline. D2 denotes

[1] https://github.com/MKT-Dataoceanai/CNVSRC2023Baseline

Table 1: WER (%) Comparison between the proposed system and the official baseline on the eval set.

System	Model	CER (%)
B1	Official baseline	48.60
M1	Proposed system	38.09

Table 2: WER (%) Ablation study of the proposed system on the valid set. The evaluation metric is the character error rate (CER%).

System	Model	Valid set 1	Valid set 2
M1D2	Proposed system	-	36.46
M1D1	Proposed system	40.46	40.37
M2D1	M1D1 - RNNLM	40.62	40.51
M3D1	M2D1 - char unit	42.36	42.38
M4D1	M3D1 - Bi-transformer decoder	43.19	43.15
M5D1	M4D1 - Inter CTC residual module	48.57	48.34

that the train set and the valid set are the same as described in Section 1. From the table, we see that the CER of the same model in valid set 1 is almost the same as the CER in valid set 2, which means that the valid set 2 is effective as valid set 1. Firstly, Inter CTC residual module and Bi-transformer decoder give a noticeable gain. Secondly, using Chinese characters as the modeling unit also improves the recognition accuracy. In addition, the RNN language model improves the performance of the system. Most importantly, the more utterances in *CNVSRC-Single.Dev* used as the train set, the more accuracy the model achieves.

4. References

[1] Chen Chen, Dong Wang, and Thomas Fang Zheng, "Cncvs: A mandarin audio-visual dataset for large vocabulary continuous visual to speech synthesis," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[2] Pingchuan Ma, Stavros Petridis, and Maja Pantic, "End-to-end audio-visual speech recognition with conformers," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 7613–7617.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[4] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al., "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[5] Jaesong Lee and Shinji Watanabe, "Intermediate loss regularization for ctc-based speech recognition," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6224–6228.

[6] Jumon Nozaki and Tatsuya Komatsu, "Relaxing the conditional independence assumption of ctc-based asr by conditioning on intermediate predictions," *arXiv preprint arXiv:2104.02724*, 2021.

[7] Maxime Burchi and Radu Timofte, "Audio-visual efficient conformer for robust speech recognition," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 2258–2267.

[8] Di Wu, Binbin Zhang, Chao Yang, Zhendong Peng, Wenjing Xia, Xiaoyu Chen, and Xin Lei, "U2++: Unified two-pass bidirectional end-to-end model for speech recognition," *arXiv preprint arXiv:2106.05642*, 2021.

[9] Binbin Zhang, Di Wu, Zhendong Peng, Xingchen Song, Zhuoyuan Yao, Hang Lv, Lei Xie, Chao Yang, Fuping Pan, and Jianwei Niu, "Wenet 2.0: More productive end-to-end speech recognition toolkit," *arXiv preprint arXiv:2203.15455*, 2022.

[10] 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*, 2006, pp. 369–376.

[11] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou, "Retinaface: Single-shot multi-level face localisation in the wild," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 5203–5212.

[12] Adrian Bulat and Georgios Tzimiropoulos, "How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks)," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1021–1030.

[13] Pingchuan Ma, Stavros Petridis, and Maja Pantic, "Visual speech recognition for multiple languages in the wild," *Nature Machine Intelligence*, vol. 4, no. 11, pp. 930–939, 2022.

[14] Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R Hershey, and Tomoki Hayashi, "Hybrid ctc/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, 2017.