2020 25th International Conference on Pattern Recognition (ICPR)
Milan, Italy, Jan 10-15, 2021

Audio-Visual Speech Recognition Using A Two-Step Feature Fusion Strategy

Hong Liu, Wanlu Xu, Bing Yang
Key Laboratory of Machine Perception, Shenzhen Graduate School, Peking University, China
Email: hongliu@pku.edu.cn, xuwanlu@pku.edu.cn, bingyang@sz.pku.edu.cn

***Abstract*—Lip-reading methods and fusion strategy are crucial for audio-visual speech recognition. In recent years, most approaches involve two separate audio and visual streams with early or late fusion strategies. Such a single-stage fusion method may fail to guarantee the integrity and representativeness of fusion information simultaneously. This paper extends a traditional single-stage fusion network to a two-step feature fusion network by adding an audio-visual early feature fusion (AV-EFF) stream to the baseline model. This method can learn the fusion information of different stages, preserving the original features as much as possible and ensuring the independence of different features. Besides, to capture long-range dependencies of video information, a non-local block is added to the feature extraction part of the visual stream (NL-Visual) to obtain the long-term spatio-temporal features. Experimental results on the two largest public datasets in English (LRW) and Mandarin (LRW-1000) demonstrate our method is superior to other state-of-the-art methods.**

***Index Terms*—speech recognition, feature fusion, non-local**

I. Introduction

Audio-visual speech recognition (AVSR) aims at combining visual information with the audio information to effectively improve the recognition accuracy in noisy environment. It has received increasing attention in recent years due to its wide applications in human-robot interaction [1], speech data mining [2], sound localization [3], etc. Common audio-visual fusion systems consist of two stages: (1) feature extraction from the image and audio signals, (2) fusion of the features for joint classification [4]. In these two stages, lots of advanced approaches have been proposed and followed. Despite significant progress, finding how to make better use of complementary information of audio and visual modalities for speech recognition is still a research focus [5].

The visual information is particularly important when the audio information is contaminated severely in a noisy environment. Hence, various researchers bend themselves to find a more effective way of lip-reading in audio-visual speech recognition tasks. Stavros *et al*. [6] presented an end-to-end visual speech recognition system based on Long-Short Memory (LSTM) networks [7] which is the first model to extract features directly from the pixels and perform classification. They then extended their approach to audio-visual fusion tasks based on residual networks and BGRU (Bidirectional Gated Recurrent Unit) in [8]. Some papers with the visual branch as key points have similar improvements [9]–[12], like replacing the full connection layer used to extract features with a 3D convolution, and then followed by standard convolutional layers or residual networks (ResNet), finally, combine LSTMs or GRUs. A common phenomenon is that most of these approaches usually extract the spatio-temporal feature of video sequences by local convolutional operation, which may lose some information between distant frames. So, how to capture the long-range dependencies of sequential data is a key point.

The way to fuse the visual and audio information is another point of audio-visual speech recognition task. There are two common kinds of fusion strategies, namely early fusion and late fusion. Early fusion is carried out in the early stage of the network, which can better preserve lossless features. While late fusion at the relatively late stage of the network, makes it better to preserve the characteristics of the different modalities themselves. Abdelaziz [14] reviewed and compared the performance of five audio-visual fusion models, and a complete evaluation of these fusion models makes his study a common benchmark in large vocabulary continuous speech recognition (LVCSR) tasks. Guo *et al*. [15] proposed a feature fusion method to combine CNN-based features and heuristic-based discriminative features that are extracted from heuristic features using deep neural network (DNN). Other papers have similar modifications [16]–[19], like changing the category or position of fusion method, designing a new loss function or improving the attention mechanism. However, most of these methods only consider the fusion in a single stage of the network, which may not be able to balance the integrity and representativeness of audio and visual information.

Considering the above research status and problems, in this paper, a non-local block [13] is inserted into the visual branch to capture long-range features of lip frames, and a two-step feature fusion strategy is proposed to combine audio and visual information in the diverse stages. The structure of the network is improved upon [8]. The main differences are the addition of a branch and the application of a non-local block. Finally, the competitive performance is obtained on LRW [20] and LRW-1000 datasets [21].

The main contributions of this paper are summarized as follows: (1) A non-local block is inserted in the feature extraction part of the visual stream (NL-Visual) to capture long-range dependencies by calculating the distance of all positions. (2) An audio-visual early feature fusion (AV-EFF) stream is added to form a two-step feature fusion strategy that can guarantee integrity and representativeness of features simultaneously. The experimental results show that our method can improve the fusion performance in strong noise environment greatly.

978-1-7281-8808-9/20/$31.00 ©2020 IEEE

2020 25th International Conference on Pattern Recognition (ICPR) | 978-1-7281-8808-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICPR48806.2021.9412454

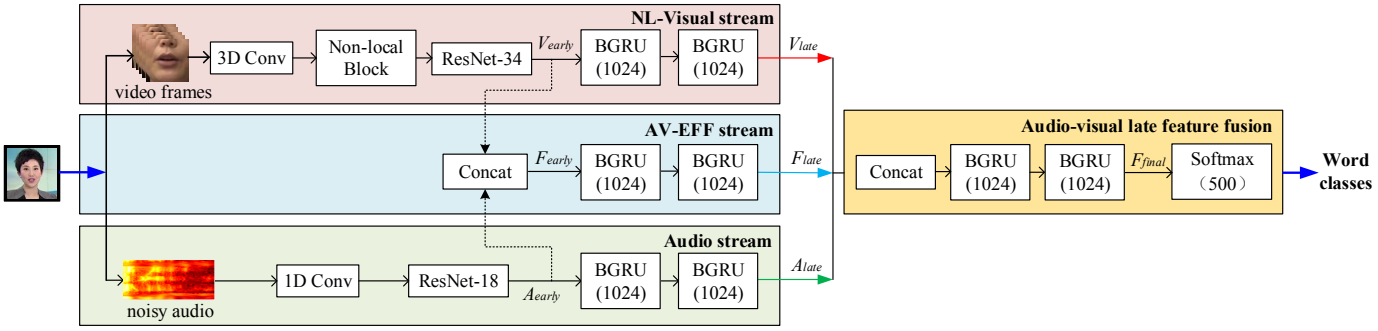

Fig. 1: Overall framework of the two-step feature fusion network, which consists of two parts. The first part has three streams including NL-Visual stream, audio stream, and audio-visual early feature fusion (AV-EFF) stream. The second part is audio-visual late feature fusion including 2-layer BGRU followed by a softmax layer that is connected with the output word label.

II. The Proposed Method

In this section, the overall framework of the two-step feature fusion network for audio-visual speech recognition is illustrated in Fig. 1. Description of variables in Fig. 1: V_{early}, A_{early} and F_{early} denote early visual, audio and audio-visual features, respectively. V_{late}, A_{late} and F_{late} denote late visual, audio and audio-visual features, respectively. F_{final} denotes the feature obtained after the late feature fusion. Our proposed network consists of three single streams and a late feature fusion part. In this section, we introduce the entire network as follows: we firstly describe GRU based NL-Visual stream and audio stream respectively. Then, the features in the middle layer of the NL-Visual and audio stream are extracted and encoded as the input of the audio-visual early feature fusion (AV-EFF) stream. After that, the outputs of the BGRU layers in three independent streams are fed to another 2-layer BGRU for late feature fusion. Finally, the loss function is mentioned.

A. NL-Visual Stream

The visual stream consists of several steps. Firstly, a 3D convolution network is used to extract the spatio-temporal information of the lip frames. Secondly, a non-local block is inserted to obtain the long-range dependencies of the entire video information. Thirdly, a 34-layer ResNet followed by a 2-layer BGRU is used for deep feature extraction, and these features will be used for subsequent early and late fusion to form a two-step feature fusion strategy. We named this non-local inserted visual stream as NL-Visual.

For the pre-processing of the video, the lip frames of each corpus regulated to 29 frames, and data augmentation technique is used to randomly flip and crop the lip region of the dataset. Then, a 3D convolutional layer [22] is used to capture the short-term dynamics of the mouth region and is proved to be advantageous. It consists of a convolutional layer with 64 3D kernels of 5 by 7 by 7 size, followed by batch normalization and rectified linear units. Formally, the output of the 3D CNN is given by:

$$v_{ij}^{xyz} = tanh\left(b_{ij} + \sum_{m}\sum_{p=0}^{P_i}\sum_{q=0}^{1}\sum_{r=0}^{Q_i}\sum_{(i\ \ 1)m}^{1 R_i 1} w_{ijm}^{pqr} v^{(x+p)(y+q)(z+r)}\right), \tag{1}$$

where the output v_{ij}^{xyz} means the value at position (x, y, z) on the jth feature map in the ith layer, where R_i is the size of the 3D kernel along the temporal dimension, w_{ijm}^{pqr} is the (p, q, r)th value of the kernel connected to the mth feature map in the previous layer.

Then, in order to capture long-range dependencies and deal with occlusion and misalignment, a spatio-temporal non-local block [13] is inserted here to compute the response at a position as a weighted sum of the features at all positions. The generic non-local operation in deep neural networks can be given by:

$$y_i = \frac{1}{\mathcal{C}(x)} \sum_{\forall j} f(x_i, x_j) g(x_j), \tag{2}$$

here i is the index of an output position whose response is to be computed, and j means index of positions. x is the input signal, and y is the output signal of the same size as x. f is a pairwise function which computes a scalar between i and all j. The unary function g computes a representation of the input signal at the position j. The response is normalized by a factor $\mathcal{C}(x)$. The difference from the convolutional operation is that non-local behavior is due to the fact that all positions $(\forall j)$ are considered in the operation.

Another question is about the choice of f and g. The experiments in [13] show that non-local models are not sensitive to these choices. So, in this paper, we only consider g in the form of a linear embedding: $g(x_j) = W_g x_j$, where W_g is a weight matrix to be learned. And for the pairwise function f, we choose embedded gaussian function:

$$f(x_i, x_j) = e^{\theta(x_i)^T \phi(x_j)}, \tag{3}$$

here we set $\mathcal{C}(x) = \sum_{\forall j} f(x_i, x_j)$ in (2). $\theta(x_i) = W_\theta x_i$ and $\phi(x_j) = W_\phi x_j$ are two embeddings.

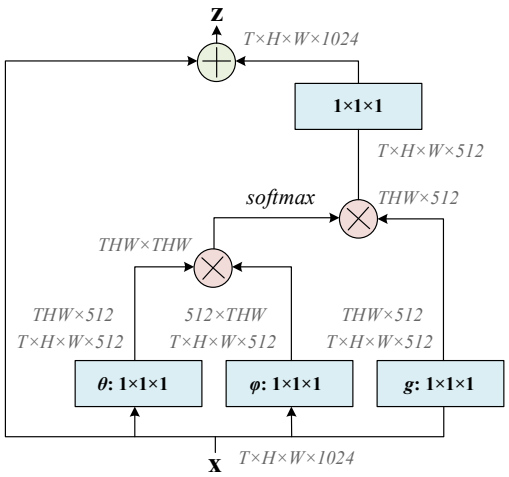

Fig. 2: A space-time non-local block. The feature maps are shown as the shape of their tensors. The shape of the input X is T(batchsize)$\times H$(height)$\times W$(width)$\times$1024(channel). "$\otimes$" denotes matrix multiplication, and "$\oplus$" denotes element-wise sum. The softmax operation is performed on each row. The blue boxes denote $1\times1\times1$ convolutions. Here is the embedded Gaussian version, with a bottleneck of 512 channels.

The non-local operation in (2) can form a non-local block which can be incorporated into many existing architectures. The definition of a non-local block is:

$$z_i = W_z y_i + x_i, \tag{4}$$

where y_i is given in (2) and "$+x_i$" denotes a residual connection [23]. The residual connection allows the non-local block to insert into any pre-trained model, without breaking its initial behavior (*e.g.*, if W_z is initialized as zero). An example space-time non-local block is illustrated in Fig. 2.

After taking Eq. (2), (3) into (4) and taking the output of 3D CNN: v as input, we can get the output of non-local block:

$$Out_{nl} = W_z \frac{1}{\sum_{\forall j} f(x_i, x_j)} \sum_{\forall j} e^{W_\theta v_i^T W_\phi v_j} W_g v_j + v_i. \tag{5}$$

Then, a 34-layer ResNet which is proposed for ImageNet [24] is followed, and the feature extracted here is called early visual feature V_{early}. After the feature transform through another two-layer BGRU, the late visual feature V_{late} is obtained. These two features can be formulated as:

$$\begin{aligned} V_{early} &= ResNet34(Out_{nl}), \\ V_{late} &= BGRU(V_{early}), \end{aligned} \tag{6}$$

where $ResNet34(\cdot)$ denotes the feature extraction operation through a ResNet-34 network [23], and the specific structure of this network is shown in Table I. $BGRU(\cdot)$ denotes the feature transform through a two-layer BGRU. It is composed of forward GRU and backward GRU, and the resulting tensor from them is spliced as the final BGRU output. GRU structure is shown in Fig. 3, which is an evolution of the LSTM, but it differs from the LSTM in many ways. Firstly, GRU removes the cell state in LSTM and only using a hidden state. Secondly, the update gate in GRU is used to replace the input gate and forgotten gate in LSTM. Thirdly, the output gate in LSTM is canceled, the reset gate is added. The advantages of GRU are that GRU can achieve similar performance to LSTM with smaller parameters, lower training costs, and faster speed.

TABLE I: Architectures for ResNet-18, ResNet-34. Building blocks are shown in brackets, with the numbers of blocks stacked. Downsampling is performed by conv3_1, conv4_1, and conv5_1 with a stride of 2.

layer name	output size	18-layer	34-layer
conv1	112×112	7×7, 64, stride 2	
conv2_x	56×56	3×3 max pool, stride 2	
		$\begin{bmatrix} 3\times3, & 64 \\ 3\times3, & 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, & 64 \\ 3\times3, & 64 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, & 128 \\ 3\times3, & 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, & 128 \\ 3\times3 & 128 \end{bmatrix}\times4$
conv4_x	14×14	$\begin{bmatrix} 3\times3, & 256 \\ 3\times3, & 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, & 256 \\ 3\times3, & 256 \end{bmatrix}\times6$
conv5_x	7×7	$\begin{bmatrix} 3\times3, & 512 \\ 3\times3, & 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, & 512 \\ 3\times3, & 512 \end{bmatrix}\times3$
out_layer	1×1	average pool, 1000-d fc, softmax	

B. Audio Stream

The audio stream architecture is similar to the visual stream. It can be summarized as the following steps. Firstly, a 1D convolution network is used to extract the audio features. Secondly, an 18-layer ResNet followed by a 2-layer BGRU is used for deeper feature extraction. Similarly, some of the features extracted in the audio stream process will be used in the later two-step feature fusion strategy.

In the pre-processing part, the speech signal $s(n)$ is pre-weighted, framed and windowed to obtain the time domain signal $x(n)$. And then take the fast fourier transform (FFT) of $x(n)$ to get the linear spectrum of $x(k)$:

$$x(k) = \sum_{n=0}^{N-1} x(n) e^{\frac{j2\pi nk}{N}}, \quad 0 \le n, K \le n-1. \tag{7}$$

After getting the input spectrum, it is important to mention that data augmentation algorithm is also used in audio stream. When the audio spectrum is fed into the network, -5 to 20 dB babble noise will be added randomly to enhance the robustness of the network architecture to the noise. Then in order to extract fine-scale spectral information from one-dimensional sound signal, 1D convolution with a temporal kernel of 5ms and a stride length of 0.25ms is used here. The output can be expressed as:

$$a_{ij}^x = tanh\left(b_{ij} + \sum_m \sum_{p=0}^{P_i-1} w_{ijm}^p a_{(i-1)m}^{(x+p)}\right), \tag{8}$$

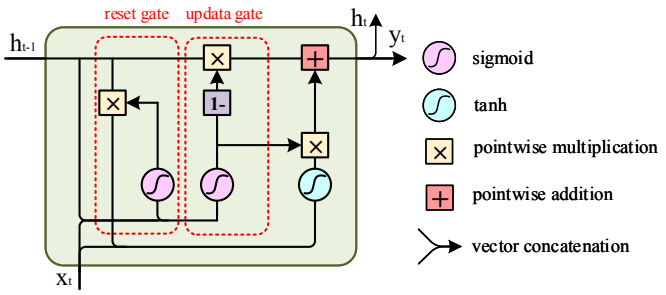

Fig. 3: GRU structure diagram. x_t is the input feature, y_t is the output feature, h_{t-1} is the hidden state generated at last time t-1, h_t is the hidden state generated at current time t.

The output a_{ij}^x means the value at position x on the jth feature map in the ith layer, where P_i is the size of the 1D temporal kernel, w_{ijm}^p is the pth value of the kernel connected to the mth feature map in the previous layer.

Since the audio signal is one-dimensional, we do not add a non-local block to the audio stream. And similar to the visual stream, a ResNet-18 network is used to extract deeper sound feature which named early audio feature A_{early}. The output of the ResNet is divided into 29 frames using average pooling to ensure the same frame rate as the video. At last, the output of the ResNet-18 is fed to a 2-layer BGRU to obtain late audio feature A_{late}. These two features can be represented as:

$$\begin{aligned} A_{early} &= ResNet18(a), \\ A_{late} &= BGRU(A_{early}), \end{aligned} \tag{9}$$

where a is the output of the 1D convolution. $ResNet18(\cdot)$ denotes the feature extraction operation through a ResNet-18 network [23], and the structure is shown in Table I. $BGRU(\cdot)$ is mentioned above.

C. Audio-Visual Early Feature Fusion Stream

The audio-visual early feature fusion is called AV-EFF for short, which is the first step in our two-step feature fusion strategy. It mainly uses the features of the middle layer in visual and audio streams for fusion. It includes the initial convolution network, deeper ResNet network, and backend BGRU layers. The details of each step are described below.

The preprocessing method of image and speech is the same as that of a single stream. A 3D convolution network connects a non-local block is used to extract the visual features, a 1D convolution network is used to extract the audio features. And then the visual and audio features go through ResNet-34 and ResNet-18 respectively, a concatenation operation is used to obtain the early audio-visual feature F_{early}. Through the other 2-layer BGRU, late audio-visual feature F_{late} is obtained:

$$\begin{aligned} F_{early} &= Concat(V_{early}, A_{early}), \\ F_{late} &= BGRU(F_{early}), \end{aligned} \tag{10}$$

where V_{early} and A_{early} denote the early visual feature and early audio feature just like these two symbols in the single visual and single audio streams. $Concat(\cdot)$ represents a join operation of vectors. Because it is the fusion of early features from two separate branches, we named this stream as an audio-visual early feature fusion (AV-EFF) stream.

So far, the structures of three single streams are all introduced. It is not difficult to see that each stream can be a separate classifier. We will discuss their performance in the experimental section. Next, we hope to integrate the three streams into a late feature fusion, so that the visual and audio information can compensate each other, and finally achieve the robustness of our network in a noisy environment.

D. Audio-Visual Late Feature Fusion

The late feature fusion is another step in the two-step feature fusion strategy. The input used in this part is a concatenation of three single streams' BGRU outputs, these outputs are in the late stage of three single streams and complementary. As shown in Fig. 1, the concatenation feature is fed to another 2-layer BGRU to fuse the information from the audio, visual and AV-EFF streams and jointly model their temporal dynamics. the feature obtained here is F_{final}. The formula can be expressed as:

$$F_{final} = BGRU(Concat(V_{late}, A_{late}, F_{late})), \tag{11}$$

$Concat(\cdot)$ represents a join operation of $V_{late}, A_{late}, F_{late}$. $BGRU(\cdot)$ is mentioned above.

The output layer is a softmax layer followed an argmax function which provides a label to each frame. The sequence is labeled based on the highest average probability. The final fusion classification result can be obtained:

$$\begin{aligned} L_{final} &= \arg\max(softmax(F_{final})) \\ &= \arg\max_{j \in 1,\ldots,K}\left(\frac{e^{F_{final}^j}}{\sum_{k=1}^{K} e^{F_{final}^k}}\right), \end{aligned} \tag{12}$$

where L_{final} denotes the recognition category of two-step feature fusion network. $softmax(\cdot)$ is used to convert the elements in vector F_{final} to probabilities, and the sum of these probabilities is 1. $\arg\max(\cdot)$ represents an operation that takes the index of the largest value in an array.

Since the fusion location of AV-EFF stream is in the relatively early stage of the network, while the location of late feature fusion is in the relatively late stage of the network, and the training of AV-EFF stream is also earlier than the late feature fusion. So, we call this fusion method as a two-step feature fusion strategy.

E. Loss Function

Whether in the single-stream training or the late feature fusion training, we use the cross-entropy as the loss function:

$$L(y, l) = -log\frac{e^{y_l}}{\sum_{i=1}^{C} e^{y_i}}, \tag{13}$$

where y denotes the prediction, l denotes the true label. We obtain the optimal parameters of the model by minimizing the cross entropy function.

Fig. 4: Samples in LRW (left) and LRW-1000 (right) datasets.

III. EXPERIMENTS AND DISCUSSIONS

In this section, we first introduce the LRW and LRW-1000 datasets. Then, the experimental setting and training process are described in order. Finally, we present the results and analysis of three experiments: Comparisons with the state of the art methods, Ablation study, and Evaluation of two-step feature fusion method.

A. Datasets

LRW dataset Lip Reading in the Wild (LRW) dataset [20] was released in 2016, which is the largest publicly available lipreading dataset in English. The dataset consists of short segments (1.16 seconds) from BBC programs, mainly news and talk shows. It is a very challenging dataset with more than 1000 speakers, 500 words, 538766 samples, and large variation in head pose and illumination. Some example frames in the LRW dataset are shown in Fig. 4 (left).

LRW-1000 dataset LRW-1000 dataset [21] was released in 2019, which is a more challenging Naturally-Distributed Large-Scale dataset in Mandarin and contains 1000 classes with 718018 samples from more than 2000 individual speakers. Each class corresponds to the syllables of a Mandarin word composed of one or several Chinese characters. It is currently the largest word-level lipreading dataset and also the only public large-scale Mandarin lipreading dataset. This dataset aims at covering a natural variability over different speech modes and imaging conditions to incorporate challenges encountered in practical applications. Some example frames in the LRW-1000 dataset are shown in Fig. 4 (right).

Compare to many typical datasets [25]–[28] which contain less than 50 words in lip-reading, The number of words in LRW and LRWL-1000 far exceeds these datasets. This is further proof of their difficulty.

Our lip-reading and AVSR experiments in a clean environment are carried out on both two datasets. And to further verify the robustness of our method in a noisy environment, we also carry out AVSR experiment under various SNR conditions on the LRW dataset.

B. Experimental Setting

Preprocessing For visual input, the first step is to extract the mouth region of interest (ROI). The size of every ROI is 96×96 and each corpus is sampled to get 29 frames. When the ROI is fed into the network, it is randomly truncated to the size of 88×88. These steps generate an input with (32, 29, 88, 88)/(batch, frames, width, heights) dimension to the visual branch. For audio input, we make the mean of each segment zero and the standard deviation one to account for variations in different levels of loudness between the speakers.

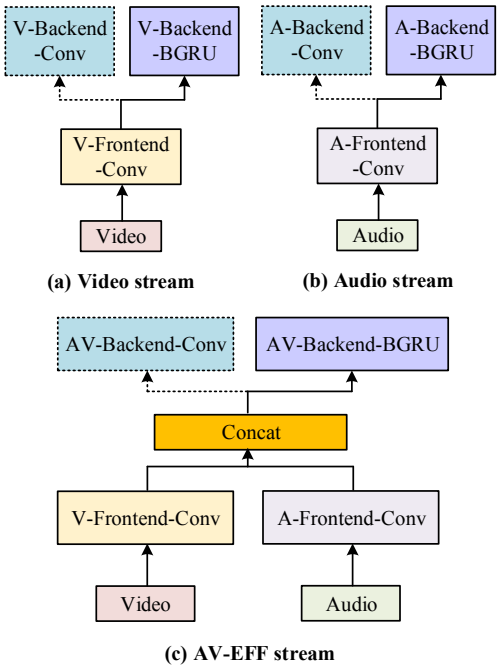

Fig. 5: The network partition of three streams in a two-step feature fusion network. The solid line represents the final classification network structure, and the dotted line represents the feature extraction network structure during the training.

Dataset partitioning The video segments in datasets are already partitioned into three sets. In the LRW dataset, there are 488766, 25000, and 25000 samples in the training, validation, and test sets. In the LRW-1000 dataset, there are 590700, 62056, and 50307 samples, respectively.

Implementation details The implementation is derived from Pytorch toolbox based on NVIDIA GeForce GTX 1080 GPU. For three single streams and the late feature fusion, the layers of BGRU are set as two, and each BGRU layer is composed of 1024 GRU neurons. Although they have the same structure, the training parameters are not shared. The number of neurons of the FC layer is equal to the number of word classes. Adam [29] is adapted to train all the networks, and the initial learning rate is set as 0.001, decreasing with the increasing of iteration times. The network is trained until there is no improvement in the classification rate on the validation set for more than 5 epochs.

C. Training Process

Training is divided into 3 phases: firstly, the audio, visual, AV-EFF streams are trained independently. And then the late

feature fusion part is trained separately with other weights fixed. Finally, the total network is finetuned end-to-end.

Network partitioning For the single stream, each network can be divided into three parts, which called frontend convolution network, backend convolution network and backend BGRU layers, shown in Fig. 5. The frontend convolution network includes early feature extraction from the convolution network to ResNet. The backend convolution network is built to connect the output of the front convolutional network to form a classifier during the feature extraction, which makes the extracted features more representational. It consists of two convolutional layers with $2\times(inputdim)$ and $4\times(inputdim)$ 1D kernels of 5 size respectively, followed by batch normalization, rectified linear units and max-pooling layer. After the training of the feature extraction network is completed. The backend convolution is discarded, and the backend BGRU layers are connected for the following training. All of the single branch networks have these parts, with only internal structural differences.

Single stream training Firstly, the frontend convolution network is connected to the backend convolution network for pre-training. After that, the backend convolution network is discarded and the backend BGRU layers are added. The backend BGRU layers are firstly trained separately with the remaining parameters fixed, and then trained end-to-end with the frontend convolution network. Early stopping is applied with a delay of 5 epochs.

Muliti stream training Once the single stream has been trained, then they are used for initializing the corresponding streams in the multi-stream architecture. Specifically, another 2-layer BGRU is added on top of all streams to fuse the single-stream outputs. The top BGRU is trained with the weights of the single-stream fixed firstly. After that, the entire network is finetuned end to end. Early stopping is also applied with a delay of 5 epochs.

D. Experimental Results

Comparisons with the state-of-the-art methods Firstly, we compare the performance of our methods with other state-of-the-art methods on LRW and LRW-1000 datasets. Since the contributions of our work lie in the visual branch and fusion method, we only show the performance of state-of-the-art lip-reading and AVSR methods in Table II. These experiments are conducted in a noiseless environment.

For lip-reading experiments on the LRW dataset, as shown in Table II, we can find that the performance of our method is superior to other state-of-the-art methods, and can achieve the best performance among them by adding non-local block to the baseline ResNet34+BGRU model [8]. On a more challenging LRW-1000 dataset, our method is better than most state-of-the-art methods, except for one [21]. The detailed comparison with the baseline method [8] is present in Ablation study.

For AVSR experiments, there are relatively fewer audio-visual results on the LRW and LRW-1000 datasets. MCNN [33] is a method based on a multidimensional convolutional neural network, which does not involve the structure of BGRU. For the fairness of comparison, we replace the front part before fusion in MCNN with our network structure. From the results in the Table II, we can find that our AVSR method surpasses other methods in a clean environment on both datasets. For example, compared to the baseline model [8], our method improved by 0.66% on the LRW dataset and 4.05% on the LRW-1000 dataset.

TABLE II: Comparison of our methods with the state-of-the-art methods on LRW and LRW-1000 datasets. Clean represents in a noiseless environment.

Task	Method	LRW Accuracy(%)	LRW1000 Accuracy(%)
Lip-reading	LSTM-5 [30]	71.50	25.76
	D3D [31]	78.02	34.76
	3D+2D [21]	83.00	**38.19**
	Multi-Grained [32]	83.34	36.91
	ResNet34+BGRU(Baseline) [8]	82.80	36.72
	NL-Visual(Ours)	**83.41**	37.03
AVSR (clean)	MCNN [33]	96.98	39.60
	ETE-AVSR(Baseline) [8]	97.60	37.52
	Two-Step(Ours)	**98.26**	**41.57**

Ablation study To further investigate the robustness of our AVSR method to noise and explore the contributions of various parts of our method to performance booming, we run ablation experiments under varying noise levels on the LRW dataset, and the results are shown in Table III. The audio signal for each sequence is corrupted by additive babble noise from the NOISEX database [34], the SNR varies from -5 to 20 dB.

For Baseline+NL-Visual, the performance improves at each SNR compared to the baseline [8]. Although the improvement seems slight in a clean environment, it is more obvious in a noisy environment, especially at -5 dB, where the performance increased by 1.55%. That proves the added non-local block plays a role. A major reason for the performance booming is that non-local block can obtain relevant information over long distances. In the case of similar lip shape changes, sometimes adjacent local information can no longer represent the features of the whole word, then distant lip frames may bring more representational features. The compensation of lip information is more obvious at low SNR condition, which makes the improvement of AVSR performance more significant in the strong noise environment.

For Baseline+AV-EFF, the performance improvements are obvious under all SNR conditions. For example, the performance of our method is 0.24% higher than the baseline in a clean environment, and 3.99% higher at -5 dB. That proves the effectiveness of AV-EFF. Compared to the baseline model using only late feature fusion, the added AV-EFF combines visual and audio features in the early stages of the network, and captures more integrated features, so that our network can learn more hidden audio-visual information.

For Baseline+NL-Visual+AV-EFF, which is our proposed two-step feature fusion strategy, it can achieve higher performance than the cases mentioned above. The improvement of accuracy increases with the decrease of SNR. In a clean

TABLE III: Ablation experiments of our two-step feature fusion method under different SNR(dB) conditions.

Baseline [8]	NL-Visual	AV-EFF	-5	0	5	10	15	20	clean
✓			86.66	94.13	96.29	96.70	97.00	97.50	97.90
✓	✓		88.21	95.01	97.18	97.22	97.53	97.86	98.10
✓		✓	90.65	95.56	97.28	97.74	98.04	98.08	98.14
✓	✓	✓	**92.10**	**96.19**	**97.35**	**97.86**	**98.08**	**98.15**	**98.26**

TABLE IV: Performance of our three single streams and fusion model under different SNR(dB) conditions.

Modality	Method	-5	0	5	10	15	20	clean
Single	Audio only	71.60	90.55	95.34	96.89	97.32	97.58	97.70
	Visual only	83.41	83.41	83.41	83.41	83.41	83.41	83.41
	AV-EFF only	87.63	94.68	96.19	96.69	96.96	97.02	97.10
Fusion	Two-step(Ours)	**92.10**	**96.19**	**97.35**	**97.86**	**98.08**	**98.15**	**98.26**

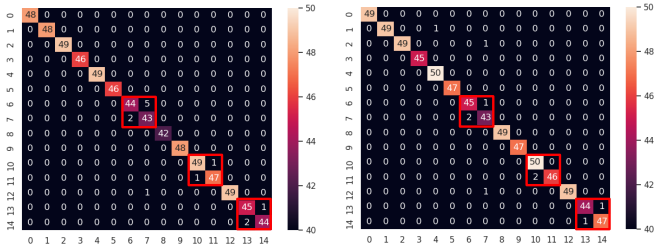

Fig. 6: Confusion matrices of baseline model (left) and our two-step feature fusion network (right) at -5dB SNR.

environment, our method is 0.36% higher than the baseline model, 0.16% higher than the NL-Visual added model, and 0.12% higher than the AV-EFF added model. At -5 dB, our method is 5.44%, 3.89%, and 1.45% higher separately. It demonstrates that our two-step feature fusion strategy can capture more information from different stages of the network than the single-stage fusion strategy, which makes our AVSR method bring performance booming in all SNR conditions.

Fig. 6 is the confusion matrices of the baseline model and our two-step feature fusion network at -5dB. We compare 15 words that begin with the letter A. The digits in confusion matrices represent the number of samples classified into each category, with the correct number on the diagonal. We can find that the confused discriminant pairs always appear on both adjacent sides of the diagonal. This is because the sample input order in the test set is alphabetized, which makes the adjacent words have a high similarity. The red boxes in the confusion matrices indicate three pairs of words that are seriously confused: ACTION and ACTUALLY, AFRICA and AFTER, AGAIN, and AGAINST. These pairs have lots of same letters and sound very similar. In these difficult pairs, the comparison between two confusion matrices verifies the superiority of our method.

Evaluation of two-step feature fusion method To demonstrate the effectiveness of our proposed two-step feature fusion method, we compare it with single-modality methods and other fusion methods under different SNR conditions.

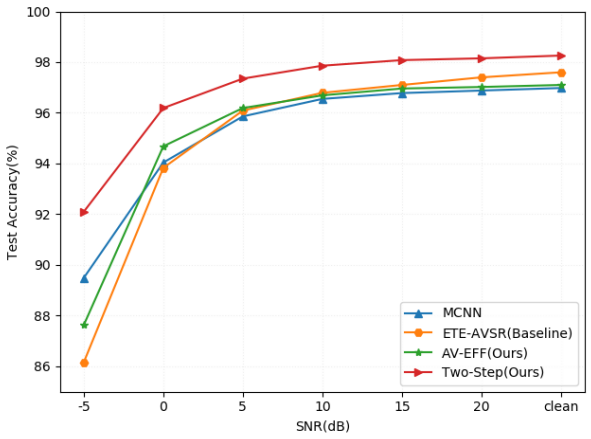

Fig. 7: Classification accuracy of different fusion methods under different SNR(dB). MCNN represents decision fusion. ETE-AVSR represents late feature fusion of the baseline model [8]. AV-EFF represents early feature fusion. Two-Step represents our two-step feature fusion.

For the comparison of different modalities, as shown in Table IV, we can find that our method has a significant improvement over single modality. For example, the performance of our fusion method is 0.56% higher than the audio-only modality in a clean environment, and at -5dB, the increase goes up to 20.5%. That demonstrates our fusion method improves obviously under the environment of strong noise, and proves the necessity of using visual information to compensate for the contaminated audio information in a noisy environment.

The comparison of different fusion methods is shown in Fig. 7. The first method [33] is decision fusion, which directly accumulates the output results of the classification layer of visual and audio streams, and then obtains the final category through an argmax function. The second method [8] is our baseline, which uses a single late-stage feature fusion strategy to ensure the independence of the audio and visual features. The third one is the AV-EFF branch that we added, which contains an early-stage feature fusion to extract more integrated features. The performance of our two-step feature

fusion strategy is superior to the other three models at each SNR. It's worth noting that our added AV-EFF's performance is higher than the baseline model when the SNR is lower than 10dB, but lower when the SNR is higher than 10dB. The reason it doesn't work well when the SNR is high is that early feature fusion destroys the independent advantage of the features to some extent. And that's why we don't make decisions directly using AV-EFF stream, but instead use a combination of early and late feature fusion.

IV. Conclusion

This paper presents a two-step feature fusion network for audio-visual speech recognition. For the visual stream, a non-local block is inserted to capture long-range dependencies among the sequential lip frames. For the fusion method, a two-step feature fusion strategy is proposed to capture the features of different stages while ensuring the integrity and independence of the features. This strategy consists of an audio-visual early feature fusion (AV-EFF) stream which can obtain more integrated features in early stage, and a late feature fusion part which can preserve the properties of different features. Experimental results on LRW and LRW-1000 datasets show the effectiveness of the non-local block and our fusion strategy, demonstrate our proposed method is superior to other state-of-the-art methods, especially in the environment of strong noise.

Acknowledgment

This work is supported by National Natural Science Foundation of China (No.61673030,U1613209), National Natural Science Foundation of Shenzhen (No.JCYJ20190808182209321).

References

[1] H. Liu, T. Fan, and P. Wu, "Audio-visual keyword spotting based on adaptive decision fusion under noisy conditions for human-robot interaction," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 6644–6651.

[2] S. Taylor, A. Kato, I. A. Matthews, and B. P. Milner, "Audio-to-visual speech conversion using deep neural networks." in *Interspeech*, 2016, pp. 1482–1486.

[3] C. Pang, H. Liu, J. Zhang, and X. Li, "Binaural sound localization based on reverberation weighting and generalized parametric mapping," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 8, pp. 1618–1632, 2017.

[4] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A. W. Senior, "Recent advances in the automatic recognition of audiovisual speech," *Proceedings of the IEEE*, vol. 91, no. 9, pp. 1306–1326, 2003.

[5] A. K. Katsaggelos, S. Bahaadini, and R. Molina, "Audio-visual fusion: Challenges and new approaches," *Proceedings of the IEEE*, vol. 103, no. 9, pp. 1635–1653, 2015.

[6] S. Petridis, Z. Li, and M. Pantic, "End-to-end visual speech recognition with lstms," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2592–2596.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] S. Petridis, T. Stafylakis, P. Ma, F. Cai, G. Tzimiropoulos, and M. Pantic, "End-to-end audio-visual speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 6548–6552.

[9] M. Wand, J. Koutník, and J. Schmidhuber, "Lipreading with long short-term memory," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 6115–6119.

[10] Y. M. Assael, B. Shillingford, S. Whiteson, and N. De Freitas, "Lipnet: End-to-end sentence-level lipreading," *arXiv preprint arXiv:1611.01599*, 2016.

[11] T. Stafylakis and G. Tzimiropoulos, "Combining residual networks with lstms for lipreading," *arXiv preprint arXiv:1703.04105*, 2017.

[12] T. Afouras, J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman, "Deep audio-visual speech recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[13] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7794–7803.

[14] A. H. Abdelaziz, "Comparing fusion models for dnn-based audiovisual continuous speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 3, pp. 475–484, 2017.

[15] L. Guo, L. Wang, J. Dang, L. Zhang, and H. Guan, "A feature fusion method based on extreme learning machine for speech emotion recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2666–2670.

[16] Y. Yuan, C. Tian, and X. Lu, "Auxiliary loss multimodal gru model in audio-visual speech recognition," *IEEE Access*, vol. 6, pp. 5573–5583, 2018.

[17] M. Wand, J. Schmidhuber, and N. T. Vu, "Investigations on end-to-end audiovisual fusion," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 3041–3045.

[18] P. Zhou, W. Yang, W. Chen, Y. Wang, and J. Jia, "Modality attention for end-to-end audio-visual speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6565–6569.

[19] V. Estellers, M. Gurban, and J.-P. Thiran, "On dynamic stream weighting for audio-visual speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 4, pp. 1145–1157, 2011.

[20] J. S. Chung and A. Zisserman, "Lip reading in the wild," in *Asian Conference on Computer Vision*, 2016, pp. 87–103.

[21] S. Yang, Y. Zhang, D. Feng, M. Yang, C. Wang, J. Xiao, K. Long, S. Shan, and X. Chen, "LRW-1000: A naturally-distributed large-scale benchmark for lip reading in the wild," in *IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, 2019, pp. 1–8.

[22] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2012.

[23] K. He, X. Zhang, and S. Ren, "Deep residual learning for image recognition," in *IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European Conference on Computer Vision*, 2016, pp. 630–645.

[25] E. K. Patterson, S. Gurbuz, Z. Tufekci, and J. N. Gowdy, "Moving-talker, speaker-independent feature study, and baseline results using the cuave multimodal speech corpus," *EURASIP Journal on Advances in Signal Processing*, vol. 2002, no. 11, p. 208541, 2002.

[26] M. Cooke, J. Barker, S. Cunningham, and X. Shao, "An audio-visual corpus for speech perception and automatic speech recognition," *The Journal of the Acoustical Society of America*, vol. 120, no. 5, pp. 2421–2424, 2006.

[27] I. Anina, Z. Zhou, G. Zhao, and M. Pietikäinen, "Ouluvs2: A multi-view audiovisual database for non-rigid mouth motion analysis," in *IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, vol. 1, 2015, pp. 1–5.

[28] G. Zhao, M. Barnard, and M. Pietikainen, "Lipreading with local spatiotemporal descriptors," *IEEE Transactions on Multimedia*, vol. 11, no. 7, pp. 1254–1265, 2009.

[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[30] J. S. Chung and A. Zisserman, "Learning to lip read words by watching videos," *Computer Vision and Image Understanding*, vol. 173, pp. 76–85, 2018.

[31] B. Shillingford, S. Whiteson, and N. d. F. Y. Assael, "Lipnet: Sentence-level lipreading," in *GPU Technology Conference*, 2016.

[32] C. Wang, "Multi-grained spatio-temporal modeling for lip-reading," *arXiv preprint arXiv:1908.11618*, 2019.

[33] R. Ding, C. Pang, and H. Liu, "Audio-visual keyword spotting based on multidimensional convolutional neural network," pp. 4138–4142, 2018.

[34] A. Varga and H. J. M. Steeneken, "Assessment for automatic speech recognition ii: Noisex-92: a database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Communication*, vol. 12, no. 3, pp. 247–251, 1993.