Signal, Image and Video Processing (2020) 14:981–989
https://doi.org/10.1007/s11760-019-01630-1

ORIGINAL PAPER

Lipreading with DenseNet and resBi-LSTM

Xuejuan Chen[1] · Jixiang Du[1] · Hongbo Zhang[1]

Received: 31 July 2019 / Revised: 8 November 2019 / Accepted: 28 December 2019 / Published online: 24 January 2020

Abstract
Lipreading is to recognize what the speakers say by the movement of lip only. Most of the previous works are to solve the problem of lipreading in English. For Mandarin lipreading, there are a few researches due to the lack of datasets. For that reason, we introduce a simple method here to build a dataset for sentence-level Mandarin lipreading from programs like news, speech and talk show. We use Hanyu Pinyin (a phonemic transcription of Chinese) as label and totally have 349 classes, while the number of Chinese characters is 1705 in our dataset. Therefore, for lipreading, there are two steps. The first step is to obtain the Hanyu Pinyin sequence. We propose a model that is composed of a 3D convolutional layer with DenseNet and residual bidirectional long short-term memory. After this, in order to get the final Chinese characters results, a model with a stack of multi-head attention is applied to convert Hanyu Pinyin into Chinese characters.

Keywords Lipreading · Sentence-level Mandarin lipreading · Visual speech recognition · Deep learning

1 Introduction

Lipreading (also called visual speech recognition) is the ability to recognize what the speakers say only relying on the visual information. It is a very challenging task for a novice because of the large number of homophones, which means that different characters are produced exactly by the same lip sequence. This phenomenon exists in many languages, such as Chinese and English. However, lipreading has a wide range of applications. Lipreading can improve anti-interference ability of audio-based speech recognition system, especially in a noisy environment [1]. It is also useful in a lot of other applications, such as helping hearing impaired person and assisting the public security alert.

Though much applications show the importance of research on lipreading, it still is a tough problem. Previous work tells that human does not perform very well in lipreading. For a limited subset of 30 monosyllabic words, people achieve an accuracy of 17 ± 12%, while 21 ± 11% for 30 compound words [2]. In another study, the lip readers who have professional experience achieve the accuracy of 26.2% on LRS dataset from BBC news, while they are allowed to watch the video as many times as they wished [3]. Since methods based on deep learning have better performance than human in many challenging works, we also try to use deep learning approaches to acquire a good performance in lipreading in this work.

Lipreading is a field that combines natural language processing with computer vision. With the development of deep learning across these two fields, lipreading approaches have been transferred from handcrafted features with HMM-based models to deep features with end-to-end architectures. Recently, the encoding–decoding lipreading system has achieved an accuracy of 50% at LRS2-BBC dataset via transformer-seq2seq architectures with external language model, only using visual information [1]. It shows that deep learning will play a greater role in the field of lipreading in further research.

There is one way to divide lipreading approaches into (i) word-level model (e.g., [4–6]) and (ii) sentence-level model (e.g., [1,3,7]). The former approach focuses on solving tasks

This work is supported by the Natural Science Foundation of China (Nos. 61673186 and 61871196), the Natural Science Foundation of Fujian Province of China (No. 2019J01082) and the Promotion Program for Young and Middle-aged Teacher in Science and Technology Research of Huaqiao University (ZQN-YX601).

✉ Jixiang Du
jxdu@hqu.edu.cn

[1] Xiamen Key Laboratory of Computer Vision and Pattern Recognition, Fujian Key Laboratory of Big Data Intelligence and Security, Department of Computer Science and Technology, Huaqiao University, Xiamen 361021, Fujian, China

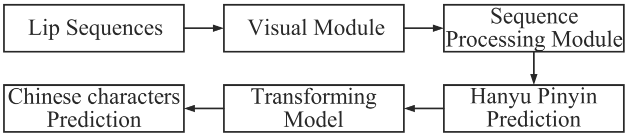

Fig. 1 The process of lipreading model

about recognition, classification, and detection of isolated word. And the latter method is to solve large vocabulary continuous speech recognition (LVCSR) and classification. As far as we know, there is no big public sentence-level Mandarin lipreading datasets for research. Therefore, we build a sentence-level dataset for Mandarin lipreading and develop an effective lipreading method.

The proposed model in this paper belongs to the latter category and it is a Mandarin lipreading system. The process of this model is shown in Fig. 1. We use the pronunciation of each word (e.g., *bo, cen, ze, ...*) as the label, and we call it as Hanyu Pinyin below. Thus, we propose two models separately to predict the Hanyu Pinyin probabilities and transform Hanyu Pinyin to Chinese characters, respectively. The former network consists of two sub-networks: (i) visual features extraction module, which applies a spatiotemporal convolution to the frame sequences and a densely connected convolutional network (DenseNet) [8] to each time step and (ii) sequence processing module, which is a two-layer residual bidirectional long short-term memory (abbreviated as resBi-LSTM) network, followed by a linear layer. And the softmax layer is applied to all time step to get the Hanyu Pinyin probabilities and the whole network is trained by CTC loss function. The latter network is composed of a stack of multi-head attention to realize the translation of Hanyu Pinyin to Chinese characters. And this model is trained by the cross-entropy loss function.

The contributions in this paper are as follows:

- We build a sentence-level Mandarin lipreading dataset, which contains 1705 Chinese characters in total.
- A novel network for sentence-level Mandarin lipreading is proposed, which contains two steps. The first step is to predict the Hanyu Pinyin probability. In this step, 3D convolution and DenseNet are applied as frontend to extracted visual features, while resBi-LSTMs are used as backend, which make use of the shallow features. After this, in the second step, a stack of multi-head attention is proposed to transform Hanyu Pinyin to Chinese characters.

This paper is arranged as follows. In Sect. 2, we review recent deep learning approaches applied on lipreading and briefly introduce the existing lipreading datasets. In order to do research on Mandarin lipreading, we build a sentence-level dataset (NSTDB) and the procedure is described in Sect. 3. After that, in Sect. 4, we introduce some useful detail information about the implementation of our model and analyze the proposed model. Finally, we show our experimental results, together with some comparative experiments and public methods in Sect. 5.

2 Related work

In this section, we outline various existing methods and major datasets on automated lipreading.

2.1 Lipreading methods

Research on automated lipreading has a long history in the computer vision fields. In the past, a large body of work on lipreading was based on the hand-designed features according to geometric information, like the lip contour. The representative methods of feature extraction were active contour models, active appearance models, active shape models and so on. At the same time, HMM was usually used backend to predict the words or sentences, and SVM classifier was also proposed to recognize the isolated word. More information about traditional lipreading methods can be thoroughly reviewed in [9] and [10].

With the success of deep learning in many fields, more and more researchers choose to use the methods in deep learning to tackle problems of extracting features and recognizing words or sentences in lipreading. A classifier CNN architecture is trained in [11] to distinguish visemes (mouth shapes), and an HMM framework is used to add temporal information after CNN output. Similarly, CNN methods combining with GMM-HMMs have been employed by [12] to extract visual features for visual speech recognition and predict phonemes in spoken Japanese. For recognizing the full word, deep bottleneck feature is applied in [13] to encode shallow input features like DCT and LDA. And similarly, in [14], a long short-term memory (LSTM) classifier has been trained with a DCT and deep bottleneck features to recognize word on OuluVS and AVLetters datasets. Meanwhile, in [4], an LSTM is used with HOG input features to recognize short phrases on GRID dataset [15].

One of deep speech recognition models is sequence-to-sequence (seq2seq) model which first reads all of the input sequence before predicting the output sentence. Seq2seq models make full use of global information of longer sequence. [3] puts forward a Watch, Listen, Attend and Spell (WLAS) sequence-to-sequence model that using an attention mechanism to both the mouth ROIs and MFCCs for continuous speech recognition. This method gets 97.0% word accuracy on GRID, while 76.2% word accuracy on LRW only using visual information. This approach is extended in

[16] to build a Multi-view Watch, Attend and Spell (MV-WAS) model for the wider variety of poses.

An end-to-end sentence-level model is first proposed by [7], which combines spatiotemporal convolutions, LSTMs and the connectionist temporal classification (CTC) loss to compute the labeling. This approach achieves 95.2% accuracy on GRID dataset. A deeper learning architecture is raised by [6], which puts forward a network including spatiotemporal convolutional, residual and Bi-LSTM networks. The goal of first two sub-networks is to extract more powerful visual features. This network attains a word accuracy of 83% on LRW. An extended version of this architecture is applied for audiovisual speech recognition [17].

Encoder–decoder architecture and CTC approaches are initially relying on recurrent networks. For example, [18] proposes a LCANet using a stacked 3D convolutional neural network (CNN), highway network, bidirectional GRU network to encode input images and a cascaded attention-CTC decoder to predict the character probabilities. This approach achieves 97.1% word accuracy on GRID dataset. Recently, some researches find that simple CNNs may perform better for sequence modeling [19]. Fully convolutional networks with CTC have been proposed for automatic speech recognition [20,21]. And also, for machine translation, [22] replaces the encoder and [23] replaces the whole pipeline with a fully convolutional network. At the same time, self-attention mechanism [24] is also found to replace recurrent networks for sequential tasks. In [1], two transformer architectures are raised, which consisted of a stack of multi-head self-attention layers with CTC and seq2seq model, respectively, and the transformer-seq2seq model combining external language model achieves the best accuracy of 50% on the LRS-BBC dataset, only using visual information.

2.2 Lipreading datasets

Some lipreading datasets before 2014 (AVICar, AVLetters, AVLetters2, OuluVS1, OuluVS2, XM2VTSDB, GRID) are reviewed in [9]. Subsequently, the datasets from BBC TV and TED (LRW, LRS, LRS2) are built by [3,5,25]. Excepting that GRID, LRS and LRS2 consist of sentences, most of the datasets only contain isolated word, and the LRS2 is the largest, but these datasets are all English. Recently, [26] present a large-scale Mandarin lipreading dataset, which named LRW-1000. To the best of our knowledge, it is the only public large-scale word-level Mandarin lipreading dataset and it offers a baseline method which achieves 38.19% accuracy.

3 News, Speech, Talk show dataset (NSTDB)

In this section, we describe the process of building a dataset for lipreading. Lipreading datasets are the most important part in lipreading research. So, we build the News, Speech, Talk show dataset (NSTDB) for Mandarin lipreading, and this dataset we collect is to do sentence-level Mandarin lipreading research. For the convenience of lipreading, we choose the programs in which speakers face up to the video camera, so that we use the TV programs such as News, Speech and Talk shows from the Internet. In this paper, the programs we choose are CCTV News and Logic Show and both of them are edited as 3-second snippet. The process of our dataset establishment is summarized in Fig. 2. And we give a brief general description of the method below.

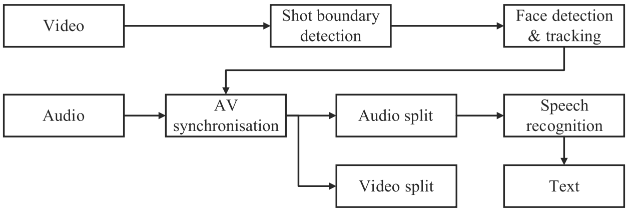

Fig. 2 Pipeline of generating News, Speech, Talk show dataset

Video preparation We first determine the shot boundaries by comparing color histograms of adjacent frames. Within each shot, we use a face detector in SeetaFaceEngine[1] to do face detection and tracking. If there is no face or the scale of face is not lager than 64 × 64 pixels, we reject the shot as not containing any potential speakers. We set the minimum measurement to filter the shots for the purpose of excluding background interference, especially many small blurred faces in the background of CCTV news. Following that, we check the video and crop the single person video in order to make it easier to get the lip pictures.

AV synchronization In the videos, we find that similar to [3,16], the video and audio streams may be out of sync for about one second. To settle this problem, we adopt the SyncNet model introduced in [27]. The model uses the two-stream CNN architecture with a contrastive loss to estimate the correlation between the mouth movement of the video and the audio track of video.

Video segmentation Some of the cropped videos are too long to train on our devices, because of the GPU memory constraints. For this reason, we choose to split the videos into 3-second video clips, as well as the audio.

Text preparation The subtitles of the TV programs may not have access to obtain. Therefore, we use the audio streams to do speech recognition to get the text via the service of Baidu Aip-Speech. The accuracy of Baidu Aip-Speech recognition is relatively high. However, there are still some errors in the alignment. Thus, we check the test sets and verification sets to ensure the correctness.

[1] SeetaFaceEngine: https://github.com/seetaface/SeetaFaceEngine

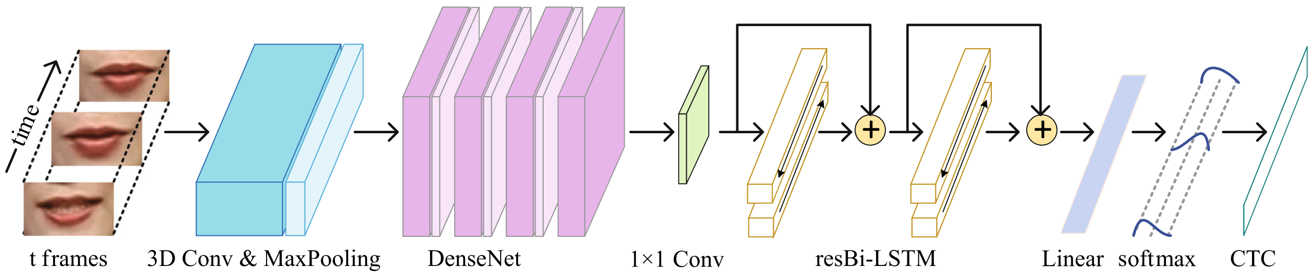

Fig. 3 The proposed model of predicting Hanyu Pinyin probabilities. A sequence of t lip frames is used as input and is proposed by a 3D convolution, followed by a max-pooling layer. Then, DenseNet is applied to further extract visual features. Next, 1 × 1 Conv is used to reduce dimension of feature. The visual features are processed by two-layer resBi-LSTM, followed by a linear layer and softmax layer. Finally, the whole network is trained by CTC loss function

Lip segmentation The videos in our dataset are the individual speech and the frame rate of videos is 25fps. In this step, we extract the mouth regions from videos. We first apply dlib [28] to detect the facial landmarks. Then, according to the landmarks, we use a mouth-centered RoI to extract lip area for each frame. Since the information of lip movement is the most important for lipreading, we only retain the lip area.

In our dataset, it contains 1705 Chinese characters in total. Some of them only appear a few times, resulting in increased difficulty of identifying. In Chinese characters, many of them have same pronunciation. For example, the pronunciation of "播" and "波" is 'bo'. So, we choose Hanyu Pinyin without tone as the label, and it only has 349 classes in our dataset.

4 The proposed methods

In this section, we describe a model architecture for lipreading. This model combines a spatiotemporal convolution and 2D convolutions to extract visual features, while two-layer resBi-LSTM is used as backend to predict the Hanyu Pinyin sequence. After this, we use a stack of multi-head attention layers to convert Hanyu Pinyin into Chinese characters.

4.1 Model of predicting Hanyu Pinyin sequence

This model structure is shown in Fig. 3.

4.1.1 Visual features extraction module

To extract visual features from lip movement, we apply a visual model combining a spatiotemporal convolution and 2D convolutions. Spatiotemporal convolutional layers can better capture time feature information in the lip sequence, while 2D convolutional layers are used to extract spatial features from input images. Therefore, we first apply a convolutional layer with 64 three-dimensional (3D) kernels of 5 × 7 × 7 size (time/height/width) on the input mouth picture sequences to obtain the spatial–temporal features, followed by batch normalization (BN) and rectified linear units (ReLU). Then, a spatiotemporal max-pooling layer is applied to cut down the spatial size of the 3D feature maps. For an input sequence of $T \times H \times W$ frames, the output is a $T \times \frac{H}{4} \times \frac{W}{4} \times 64$ tensor.

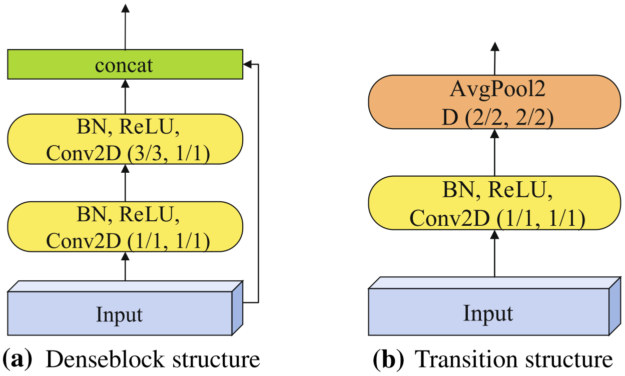

Fig. 4 Two import structures in DenseNet

After the spatiotemporal convolutional layer, a dense connection network (DenseNet) is employed to extract more spatial features at each time step and we use the 121-layer version. DenseNet can solve the problem of overfitting well because it establishes a connection between different layers to utilize shallow features with low complexity, which makes it easier to obtain a smooth decision function with better generalization performance. The DenseNet contains Denseblock layers and Transition layers. The structure of each Denseblock layer and Transition layer is separately shown in Fig. 4. In Denseblock, concat operation is used to concatenate all the outputs in previous layers as the next input. Transition layer is used between two Denseblock layers in order to reduce the number of channels which passed to the next Denseblock layer. The final BatchNorm is following after the final Denseblock layer. Finally, the output is a $T \times \frac{H}{32} \times \frac{W}{32} \times 1024$ tensor and an adaptive average pool is adopted on the spatial dimensions, yielding a 1024-dimensional vector for every input lip image.

4.1.2 Sequence processing module

A convolutional layer with 1 × 1 kernels and two-layer resBi-LSTM are applied following the visual model. The goal of the convolutional layer is to reduce the dimension from 1024 to 512. Meanwhile, a shortcut is utilized in our resBi-LSTM layer to add the original information before layer to the information after the Bi-LSTM layer. Residual structure used here allows the information extracted from visual module to be perceived by all the LSTM layers. Furthermore, this structure can learn more complicated information by integrating the viseme and semantic information, while the Bi-LSTM layers learn the semantic information independently. resBi-LSTM has been used in speech recognition, which achieved a good performance [29]. A Bi-LSTM structure is able to get the information within a sentence both forward and backward, like the word in a Chinese sentence can be logically deduced from the previous or subsequent words.

A linear layer and softmax layer are applied after the resBi-LSTM layers. Finally, the whole network is trained by CTC loss function. The CTC loss L_{ctc} is defined as follows:

$$p_{\text{ctc}}(y|x) = \sum_{w \in F^{-1}(y)} p_{\text{ctc}}(w|x) = \sum_{w \in F^{-1}(y)} \prod_{t=1}^{T} q_{w_t}^{t} \quad (1)$$

$$L_{\text{ctc}} = -\ln p_{\text{ctc}}(y|x) \quad (2)$$

where T is the length of the input sequence. $q_{(w_t)}^{t}$ indicates the softmax probability of the outputting label w_t, where $w_t \in \{a, ai, an, ao, \ldots, zun, zuo, blank\}$ at the frame t. $w = (w_1, w_2, \ldots, w_T)$ is the CTC path of a sequence and y is the sentence label (ground truth). $F^{-1}(y)$ is the set of all possible CTC paths that can be mapped to ground truth y.

4.2 Translating Hanyu Pinyin to Chinese characters

As a Mandarin lipreading task, we are supposed to get Chinese characters as the final result. Converting Hanyu Pinyin into Chinese characters is similar to machine translation, and particularly, Hanyu Pinyin and Chinese characters are equal in length. Recently, multi-head attention achieves a remarkable performance in the field of machine translation. Inspired by that, we use a stack of multi-head attention layers with feedforward blocks to train a language model to achieve the goal, followed by a linear layer. Hanyu Pinyin is embedded into a 512-dimensional vector as the input value. The information about the word order of the input sequences is fed to the model by fixed positional embedding in the form of sinusoid functions. At the same time, the input value would serve as the query, key and value. The structure of this model is shown in Fig. 5. This model is trained by cross-entropy loss function, which is described as follows:

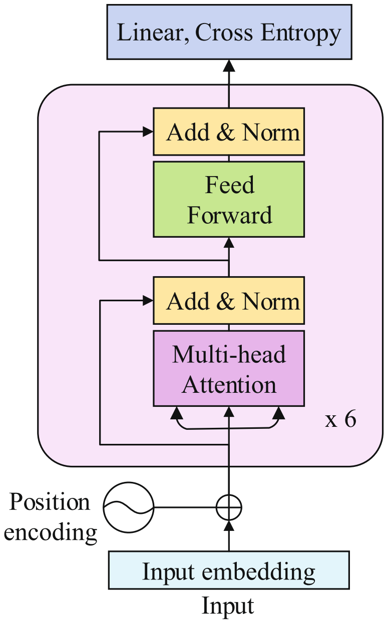

Fig. 5 The structure of translating Hanyu Pinyin to Chinese characters

$$q_i = \frac{e^{y_i}}{\sum_{j=1}^{N} e^{y_j}} \quad (3)$$

$$L = -\sum_{i=1}^{N} p_i * \ln q_i \quad (4)$$

where N is the number of classes. The ground truth is represented by p, using the form of one hot. And q is the softmax probability of prediction y. Because the value in p is 1 or 0, this loss function can be simplified as follows:

$$L = -\ln q_m \quad (5)$$

where q_m is the softmax probability of real class (ground truth).

5 Experiments

In this section, we evaluate the performance of the proposed model and compare it with the popular lipreading methods on our own NSTDB. Furthermore, we also give results on the public methods for a comparison to show the benefit of residual structure.

5.1 Implementation details and evaluation criteria

We implement and train the network using Pytorch1.0. AMS-Grad optimizer is used with an initial learning rate of 0.001, and the batch size is 16.

Table 1 Hanyu Pinyin word error rates (HPWER) using different visual modules

Method	HPWER (%)
ResNet+Bi-LSTM [6]	55.12
SqueezeNet+Bi-LSTM	62.57
VGG+Bi-LSTM	57.43
DenseNet+Bi-LSTM	53.11

Table 2 Hanyu Pinyin word error rates (HPWER) using different sequence processing modules

Method	HPWER (%)
DenseNet+Bi-LSTM	53.11
DenseNet(fc512)+resBi-LSTM	52.01
Proposed Method (DenseNet(conv)+resBi-LSTM)	50.44

Our architecture performs word-level prediction on input frames. According to the model, we use word error rate as the evaluation criteria. For the first period network, we predict the Hanyu Pinyin sequences. So, we define the Hanyu Pinyin word error rate (HPWER) as evaluation criteria,

$$\mathrm{HPWER} = \frac{\mathrm{CPS} + \mathrm{CPD} + \mathrm{CPI}}{\mathrm{CPN}} \tag{6}$$

where CPS, CPD and CPI are the number of substituted, deleted and inserted words, respectively, required to transform the Hanyu Pinyin prediction into the Hanyu Pinyin ground truth.

Similar to the HPWER, we define Chinese characters word error rate (CCWER) in the second period, which transforms Hanyu Pinyin to Chinese characters, and the formula is as follows:

$$\mathrm{CCWER} = \frac{\mathrm{CCS} + \mathrm{CCD} + \mathrm{CCI}}{\mathrm{CCN}} \tag{7}$$

5.2 Experimental results and discussion

We describe the effect of different parts in the first period to verify the effectiveness of our proposed model and analyze the reason for loss of accuracy in the process of translating Hanyu Pinyin to Chinese characters.

5.2.1 The effect of visual module

We first implement the method in [6] using Pytorch1.0. It puts forward a model combining spatiotemporal convolutional layers with 34-layer ResNet to extract powerful features, followed by a linear layer and two-layer Bi-LSTMs. The building blocks of ResNet consist of two convolutional layers with BN and ReLU, and the skip connections of addition operation are used to stimulate information propagation. In ResNet networks, an adaptive average pool is adopted on the spatial dimensions at last, yielding a 512-dimensional vector for every input lip image. The following linear layer is to reduce dimension from 512 to 215, and in the two-layer Bi-LSTM, the hidden size is 256. This network achieves a 55.12% HPWER on NSTDB.

Table 3 Hanyu Pinyin word error rates (HPWER) on public methods

Method	HPWER (%)
LipNet [7]	64.35
ResNet+Bi-LSTM [6]	55.12
Proposed method (DenseNet(conv)+resBi-LSTM)	50.44

Next, we analyze the impact of different visual features on the results. For the purpose of extracting more useful and robust features, we consider to replace the ResNet in the public model. Except the deep neural network which has been verified in [6], we select three classical convolutional neural network to replace ResNet. Other parts of the network are the same. So, the first 3D convolutional layer is before these three convolutional networks, and a linear layer with two-layer Bi-LSTMs of 256 hidden size is following the convolutional networks.

The first one is SqueezeNet. SqueezeNet is a lightweight network, and it contains eight Fire modules with one convolutional layer at the beginning and end. In this model, the beginning convolution is transformed to 3D convolution and the output is a 512-dimensional vector for every input lip image. Fire module is mainly composed of two parts: the squeeze layer and the expand layer. The mainly aim of the squeeze layer is to reduce the number of input channels, while the expand layer is used to merge the feature maps from 1×1 Conv and 3×3 Conv. It obtains a 62.57% HPWER on NSTDB.

The second convolutional network is VGGNet, and we use the 16-layer version. We only use the convolutional layers without the three linear layers in VGGNet. In convolutional layers, it adopts 3×3 kernels to capture changes in detail. An adaptive average pool is adopted to yield a 512-dimensional vector for every input lip image. On NSTDB, it gets a HPWER of 57.43%.

The final network is DenseNet. We choose the 121-layer DenseNet-BC, and for this network, it yields a 1024-dimensional vector after an adaptive average pool for every input lip image. The detail information of DenseNet is same as that is introduced in Sect. 4.1. And it achieves 53.11% HPWER on NSTDB, which is lower than the ResNet approach.

Table 4 Hanyu Pinyin word error rates (HPWER), Chinese characters word error rates (CCWER) and the difference between CCWER and HPWER (Diff. = CCWER − HPWER) on all networks

Method	HPWER (%)	CCWER (%)	Diff. (%)
LipNet [7]	64.35	73.19	8.84
ResNet+Bi-LSTM [6]	55.12	62.94	7.82
SqueezeNet+Bi-LSTM	62.57	70.45	7.88
VGG+Bi-LSTM	57.43	65.45	8.02
DenseNet+Bi-LSTM	53.11	61.18	8.07
DenseNet(fc512)+resBi-LSTM	52.01	59.84	7.83
Proposed Method (DenseNet(conv)+resBi-LSTM)	50.44	58.20	7.76

Table 5 Examples of homophones which cause increased error rate. The words blue and red are the substituted, deleted and inserted words in ground truth and prediction sequence

Chinese charactes	Hanyu Pinyin	Predicted Hanyu Pinyin	Hanyu Pinyin error	Predicted Chinese charactes	Chinese charactes error
只要聪明就会成为	zhi yao cong ming jiu hui cheng wei	zhi xiao you ming jiu hui cheng wei	2	只小有名就会成为	3
守岛卫国三十二年用无怨	shou dao wei guo san shi er nian yong wu yuan	you dao wei guo san shi er nian yan yong wu	3	有到卫国三十二年验用五	5
不断绿起来美起来	bu duan lv qi lai mei qi lai	bu duan zuo jin gai ba qi an	5	不断做进改八气案	6
是不是也讲得通啊	shi bu shi ye jiang de tong a	ri bu shi ye jiang de dong le	3	日不是也讲的动了	4
而原来的成型的商业体呢	er yuan lai de cheng xing de shang ye ti ne	yuan lai de cheng xing de shang li li de	4	原来的成形的上历历的	6

The results of these four networks are shown in Table 1. From the results, we can see that complex network can extract more powerful visual features. SqueezeNet is the simplest network in these four convolutional networks, so it cannot extract deeper features and the HPWER is 62.57%. Compared to SqueezeNet, VGGNet reduces 5.14% HPWER because of the deeper network layer. ResNet deepens the network and uses skip connection to solve the problem of degradation. Its HPWER achieves further 2.31% absolute reduction. Different from the addition in ResNet, DenseNet uses concat to make use of shallow information from former layers, and it has better performance that further reduces 2.01% HPWER. From these four experiments, we know that deeper network structure can extract more effective and powerful visual features.

5.2.2 The effect of sequence processing module

Except the frontend extracted features will affect the accuracy of results, the backend sequence processing methods also have an impact on the results. From the results of Sect. 5.2.1, the structure of DenseNet achieves the best results. Therefore, we use 3D convolutional layer and DenseNet to extract visual features.

Given the comparison of convolutional neural network, we can find that the shallow feature can play an important role in the deep layers. Thus, the information from visual module also can be propagated to the deep LSTM layers. For this reason, we apply resBi-LSTM to replace the Bi-LSTM. The output dimension of Bi-LSTM is twice the hidden size. And the residual operation acquires the dimensional consistency. So, we first attempt to modify the output dimension of the linear layer before Bi-LSTM layers from 256 to 512, and it gets 52.01% HPWER.

In [30], it puts forward that linear layer can be replaced by the convolutional layer of 1 × 1 kernels. Therefore, we use the convolutional layer to replace the linear layer before resBi-LSTM layers and it achieves the best accuracy in our experiments. It reduces the HPWER to 50.44%.

The results are shown in Table 2. Compared to Bi-LSTM, resBi-LSTM reduces 1.1% HPWER, which verifies the shallow features also useful in Bi-LSTM layers. A convolutional layer with 1 × 1 kernels is used to replace linear layer before resBi-LSTM, and it further reduces 1.57% HPWER. We think the reason of this result is that the linear layer destroys the spatial structure of the image to a certain extent, while the convolutional layer preserves.

5.2.3 Compare with other methods

In addition to the approach in [6], we also use a public model LipNet [7] which combines three layers of STCNN with a spatial max-pooling layer and two-layer BiGRUs as a baseline result on our dataset. We first use dropout which was described in [7]. However, the training loss does not decline and remain high after several epoch. We employ BatchNorm to replace dropout, and it can train normally. This method gets a worst result in this paper, which only 64.35%.

The results of public method are shown in Table 3. We can see that the method in [6] reduces 9.23% HPWER compared with LipNet, which also proves the importance of visual features in the model.

5.2.4 Results after transform Hanyu Pinyin to Chinese characters

After predicting Hanyu Pinyin sequences, they should be transformed to Chinese character sequences. In this paper, we use a stack of multi-head attention to realize the function of translation. The results are shown in Table 4. From the result, we can see that in the procedure of Hanyu Pinyin to Chinese characters will lose about 8% accuracy rate. In consideration of the result of lipreading, as shown in Table 5, we can find that, even Hanyu Pinyin is same, Chinese characters would be different due to the different contexts.

6 Conclusion

In this paper, we propose a deep learning network for Mandarin sentence-level lipreading. This network consists of two steps. The first network is to predict the Hanyu Pinyin sequence for input lip sequence and it is a combination of a three-dimensional convolutional layer, a DenseNet and two-layer resBi-LSTM. It is trained by a CTC loss function. The second network is to transform Hanyu Pinyin to Chinese characters. It is composed of a stack of multi-head attention and trained by cross-entropy loss. Compared to the accuracy of Hanyu Pinyin, the Chinese characters accuracy is reduced by about 8% through the second step. As a result, the final Chinese characters accuracy depends on the Hanyu Pinyin accuracy. We compare several networks for predicting Hanyu Pinyin sequence and verify the importance of the part of network. For Hanyu Pinyin and Chinese characters accuracy, the proposed network has a 13.91% and 14.99% absolutely improvement by LipNet [7], while an absolute 4.68% and 4.74% improvement compared to the ResNet method [6].

References

1. Afouras, T., Chung, J.S., Senior, A., Vinyals, O., Zisserman, A.: Deep audio-visual speech recognition. IEEE Trans. Pattern Anal. Mach. Intell. (2019). https://doi.org/10.1109/TPAMI.2018.2889052
2. Easton, R.D., Basala, M.: Perceptual dominance during lipreading. Percept. Psychophys. **32**(6), 562–570 (1982). https://doi.org/10.3758/bf03204211
3. Chung, J.S., Senior, A., Vinyals, O., Zisserman, A.: Lip reading sentences in the wild. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3444–3453 (2017)
4. Wand, M., Koutnik, J., Schmidhuber, J.: Lipreading with long short-term memory. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6115–6119 (2016)
5. Chung, J.S., Zisserman, A.: Lip reading in the wild. In: Asian Conference on Computer Vision, 2016. Computer Vision - ACCV, pp. 87–103 (2016)
6. Stafylakis, T., Tzimiropoulos, G.: Combining residual networks with LSTMs for lipreading. In: conference of the international speech communication association, pp. 3652–3656 (2017)
7. Assael, Y.M., Shillingford, B., Whiteson, S., De Freitas, N.: Lipnet: End-to-end sentence-level lipreading. arXiv preprint arXiv:1611.01599 (2016)
8. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708 (2017)
9. Zhou, Z., Zhao, G., Hong, X., Pietikainen, M.: A review of recent advances in visual speech decoding. Image Vis. Comput. **32**(9), 590–605 (2014). https://doi.org/10.1016/j.imavis.2014.06.004
10. Potamianos, G., Neti, C., Luettin, J., Matthews, I.: Audio-visual automatic speech recognition: an overview. Issues Vis. Audio-Vis. Speech Process. **22**, 23 (2004)
11. Koller, O., Ney, H., Bowden, R.: Deep learning of mouth shapes for sign language. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 85–91 (2015)
12. Noda, K., Yamaguchi, Y., Nakadai, K., Okuno, H.G., Ogata, T.: Lipreading using convolutional neural network. In: Fifteenth Annual Conference of the International Speech Communication Association, pp. 1149–1153 (2014)
13. Tamura, S., Ninomiya, H., Kitaoka, N., Osuga, S., Iribe, Y., Takeda, K., Hayamizu, S.: Audio-visual speech recognition using deep bottleneck features and high-performance lipreading. In: 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), pp. 575–582 (2015)
14. Petridis, S., Pantic, M.: Deep complementary bottleneck features for visual speech recognition. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2304–2308 (2016)
15. Cooke, M., Barker, J., Cunningham, S., Shao, X.: An audio-visual corpus for speech perception and automatic speech recognition. J. Acoust. Soc. Am. **120**(5), 2421–2424 (2006). https://doi.org/10.1121/1.2229005
16. Chung, J.S., Zisserman, A.: Lip Reading in Profile. In: british machine vision conference (2017)
17. Petridis, S., Stafylakis, T., Ma, P., Cai, F., Tzimiropoulos, G., Pantic, M.: End-to-end audiovisual speech recognition. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6548–6552 (2018)
18. Xu, K., Li, D., Cassimatis, N., Wang, X.: LCANet: End-to-end lipreading with cascaded attention-CTC. In: 2018 13th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2018), pp. 548–555 (2018)
19. Bai, S., Kolter, J.Z., Koltun, V.: An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:1803.01271 (2018)
20. Wang, Y., Deng, X., Pu, S., Huang, Z.: Residual convolutional CTC networks for automatic speech recognition. arXiv preprint arXiv:1702.07793 (2017)
21. Zhang, Y., Pezeshki, M., Brakel, P., Zhang, S., Laurent, C., Bengio, Y., Courville, A.C.: Towards end-to-end speech recognition with deep convolutional neural networks. In: Conference of the International Speech Communication Association, pp. 410–414 (2016)

22. Gehring, J., Auli, M., Grangier, D., Dauphin, Y.: A Convolutional encoder model for neural machine translation. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pp. 123–135 (2017)
23. Gehring, J., Auli, M., Grangier, D., Yarats, D., Dauphin, Y.N.: Convolutional sequence to sequence learning. In: Proceedings of the 34th International Conference on Machine Learning-Vol. 70, pp. 1243–1252 (2017)
24. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)
25. Afouras, T., Chung, J.S., Zisserman, A.: LRS3-TED: a large-scale dataset for visual speech recognition. arXiv preprint arXiv:1809.00496 (2018)
26. Yang, S., Zhang, Y., Feng, D., Yang, M., Wang, C., Xiao, J., Long, K., Shan, S., Chen, X.: LRW-1000: A naturally-distributed large-scale benchmark for lip reading in the wild. In: 2019 14th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2019), pp. 1–8 (2019)
27. Chung, J.S., Zisserman, A.: Out of time: automated lip sync in the wild. In: Asian conference on computer vision, pp. 251–263 (2016)
28. Kazemi, V., Sullivan, J.: One millisecond face alignment with an ensemble of regression trees. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1867–1874 (2014)
29. Zhou, X., Li, J., Zhou, X.: Cascaded CNN-resBiLSTM-CTC: An end-to-end acoustic model for speech recognition. arXiv preprint arXiv:1810.12001 (2018)
30. Lin, M., Chen, Q., Yan, S.: Network in network. Comput. Sci. (2013)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.