Speaker Recognition Based on 3DCNN-LSTM

ZhangFang Hu, XingTong Si, Yuan Luo, ShanShan Tang, Fang Jian

***Abstract*—The traditional speaker recognition method reduces the feature signal from high to low dimensions, but this often leads to some speaker information loss, resulting in a low speaker recognition rate. In response to this problem, this paper proposes a model based on the combination of a 3D convolutional neural network (3DCNN) and a long short-term memory neural network (LSTM). First, the model uses a fixed-step speech feature vector as the 3DCNN input, which converts the text-independent speaker recognition mode into a "semi-text"-related speaker recognition mode, which greatly preserves the speaker's speech features, and thus improving the difference between the characteristics of different speakers. Second, the 3D convolution kernel designed in this paper can extract the personality characteristics of speakers in different dimensions to further distinguish different speakers, connect the output signal to the LSTM network through a time series to enhance the contextual connection of the speaker's voice, and finally mark the classification output result to realize a complete speaker recognition system. The experimental results show that the model structure improves the speaker recognition rate on AISHELL-1 dataset in short-term speech compared with traditional algorithms and popular embedding features, and the system is more robust over time.**

***Index Terms*—speaker recognition, semi-text processing, 3DCNN, LSTM**

I. Introduction

SPEAKER recognition, also known as voiceprint recognition, is one of the most important components of biometric signal recognition [1]. Compared with the current popular biometrics, such as fingerprints, gestures, retinas, irises, and faces, voice is the most direct method of human communication. Additionally, the speaker's voice collection is more convenient, the cost is controllable, and the speaker's privacy can also be better protected.

The task of speaker recognition is to identify which speaker is speaking in the established speaker library. The speaker recognition method can be divided into text-dependent and text-independent speaker recognition according to whether the content of the speaker is predefined. It can also be divided into speaker verification and speaker identification according to whether the number of voices for speaker recognition is single. Its basic system framework is mainly divided into feature extraction and speaker models.

Feature extraction extracts the speaker's speech signal feature vectors, which can fully reflect individual differences and remain stable for a long time. Speaker features are divided into time-domain features and transform domain features. The common time-domain features include amplitude, energy, and average zero-crossing rate. However, these features are usually the feature vectors obtained from the speech signal directly through the filter, the processing is simple and its stability is poor, and the ability to express the speaker's identity information is weak, so it is rarely used. The transform domain refers to the vector features obtained by some transformation of the speech signal. Common transform domain features include LPC (linear prediction coefficient, LPC) [2], [3] BFCC (bark frequency cepstrum coefficient) [4], and MFCC (mel frequency cepstrum coefficient) [5]-[7]. The characteristic parameters of the transform domain can better imitate human voice characteristics, so it has stronger robustness and better stability and has been widely used [8], [9].

The traditional speaker model has a dynamic time planning algorithm based on template matching. This method is simple and real-time, but the data storage is small and the robustness is poor. The classical probability and statistics algorithm based on GMM-UBM (Gaussian mixture model general background model) has been widely used in various pattern recognition and achieved good results [10]-[12]. However, with the increase in recognition accuracy requirements, reference [13] showed that the model needs to determine more parameters, the calculation complexity is high, and the recognition time also increased accordingly. Until now, the widely used i-vector recognition algorithm [14], combined with a variety of channel compensation techniques, can well express the difference between speakers. Although a good recognition effect has been achieved, there is still a large difference between the training phase and the test phase [15], which is particularly evident in the recognition of text-independent speakers, and the ability to resist noise to the environment is weak [16].

Manuscript received July 4, 2020; revised February 6, 2021. This work was supported in part by Chongqing Research Program of Basic Research and Frontier Technology (Grant No. cstc2017jcyjAX0212), the Science and Technology Research Program of Chongqing Municipal Education Commission (Grant No. KJ1704072), National Natural Science Foundation of China Youth Fund (Grant No. 61703067).

ZhangFang Hu is a Professor of the Key Laboratory of Optical Information Sensing and Technology, School of Optoelectronic Engineering, Chongqing University Posts and Telecommunications, Chongqing 400065 China (e-mail: 495075688@qq.com).

Xingtong Si is a Master's degree candidate of the Department of Electronic Science and Technology, School of Optoelectronic Engineering, Chongqing University Posts and Telecommunications, Chongqing 400065 China (e-mail: 13140246046@163.com).

Yuan Luo is a Professor of the Key Laboratory of Optical Information Sensing and Technology, School of Optoelectronic Engineering, Chongqing University Posts and Telecommunications, Chongqing 400065 China (e-mail: 2217793866@qq.com).

Shanshan Tang is a Master's degree candidate of the Department of Electronic Science and Technology, School of Optoelectronic Engineering, Chongqing University Posts and Telecommunications, Chongqing 400065 China (e-mail: 1439789101@qq.com).

Fang Jian is a Master's degree candidate of the Department of Electronic Science and Technology, School of Optoelectronic Engineering, Chongqing University Posts and Telecommunications, Chongqing 400065 China (e-mail: 1172599506@qq.com).

Deep learning can be characterized in several different ways and can effectively solve the complex classification problems [17], [18]. Driven by the success of deep learning in the field of biosignal recognition and others [19], some early work on speaker recognition focused on how to use the bottleneck structure of neural networks to train frame-level speaker speech to identify individual features. This modelling method can effectively increase the signal-to-noise ratio of feature information and reduce the number of fitting phenomena in the training process so that the model has better generalization performance, but its bottleneck features still have more redundant features. The characterization ability is still weak. To solve the problem of feature redundancy, experts and scholars worldwide have proposed mapping sentences with different lengths to the fixed dimension of the embedding feature[20]-[25]. In 2014, reference 10 shows that each speech frame of a speaker can be input into a DNN neural network, and the last output activation value of the hidden layer can be used as the speaker's feature vector d-vector. However, this processing method is relatively simple, so Snyder proposed a time-delay neural network (time-delay neural network, TDNN) in 2015 that extracts x-vector feature vectors from speaker speech[26], and the statistical pooling layer in the network structure converts frame-level features into segment-level features, so it has better robustness than d-vector. The added delay structure can better capture the long correlation of speech features, but this does not greatly improve the phrase speaker recognition performance because the TDNN network structure does not make full use of the time factor in the environment of short speech duration, so that the speaker recognition rate may decline [27]. A compact e-vector speaker feature structure is then proposed to avoid the problem of system performance degradation caused by speaker speech length. While generating a more accurate speaker subspace, no additional memory or computing costs are added to the standard i-vector [28].

Based on the above analysis, this paper proposes a model based on the combination of a 3D convolutional neural network (3DCNN) and a long short-term memory network (LSTM) to improve the recognition rate of the speaker recognition system. First, the speaker's speech frame is superimposed and preprocessed in 40 ms steps. The purpose here is to convert the speaker's speech information into a semi-text related problem, extract the MFEC feature of the superimposed speech frame as a more robust speaker feature, and then process the speech signal to construct brand-new time-frequency domain-speech volume 3D data. Then, the 3D convolution kernel is entered to capture the deep speaker information. The pooling layer is followed by the LSTM network model to learn the context information of the speaker's voice because of its powerful temporal modeling ability [29]. The activation function selected in the training phase is PReLU with higher stability. The training criterion is the cross-entropy loss function. Compared with the previous single deep learning network model, the three-dimensional information used in this paper replaces the one-dimensional data commonly used in speaker recognition. To improve the speaker's ability to express features, the subsequent LSTM can maximize the retention of the speaker's voice information content, thereby improving the accuracy of recognition. Finally, the output result is sent to the fully connected class and softmax judgement to obtain the final classification result.

II. Methods

A. Overall system design

The 3D convolution kernel introduced in this paper can capture richer time-frequency information, and how to fully use this information is the key to improving the speaker recognition rate. The method used in this paper is to first process the "semi-text" of the speech signal preprocessing stage, use the processed three-dimensional signal as the input of the 3DCNN network, and then connect the LSTM after the 3DCNN network to learn the contextual content of the speaker's speech and the speaker model is obtained. The specific model framework is shown in Figure 1, including the speech signal, signal preprocessing module, 3DCNN module, LSTM module and speakers.

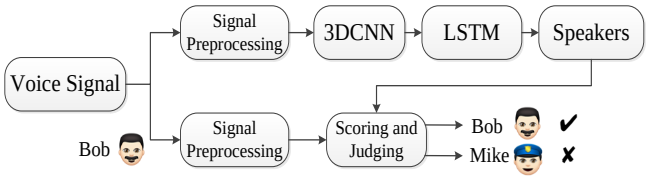

Fig. 1. Speaker recognition framework

Signal preprocessing module: It includes "semi-text" processing of the speaker's voice signal and then converts the processed voice signal into a brand-new three-dimensional data type as the input signal of the 3D convolutional neural network.

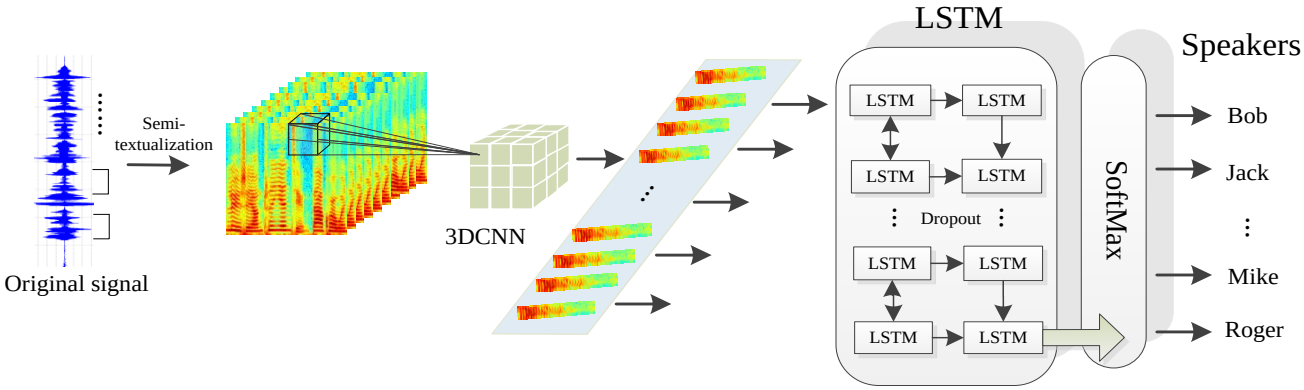

Fig. 2. 3DCNN-LSTM speaker recognition framework

3DCNN module: This article designs a brand-new three-dimensional convolution kernel, which has a stronger ability to recognize speaker features. The main function is to extract deep-level features from the input data.

LSTM module: This module uses the long-term characteristics of the speaker's voice signal to sequence the output of the convolutional neural network to learn the contextual content of the speaker's voice and finally combines softmax for identification and classification.

The schematic block diagram of the model in Figure 2 gives a more vivid model structure. The original speech signal is "semi-text" processed to obtain three-dimensional speaker voice information, and then the serialization features of speakers are obtained by using the newly designed 3D convolution kernel. The speaker feature vector is used as the input signal of the LSTM network model, and finally, the specific speaker is determined by softmax.

B. *Design signal preprocessing module*

In this paper, the logarithmic energy pair is used to process the speaker's speech signal, and the DCT operation in the extraction of traditional MFCC features is abandoned. The reason is that the DCT operation is based on the signal processing of the local operation, which contradicts the local calculation of the subsequent convolution operation [30]. It will disturb the personality characteristics of the speaker information and finally cause the ability to recognize the characteristics of the speaker to decline. Therefore, the feature signal MFEC is finally extracted by discarding the DCT operation. In this paper, a 40 ms speech window is used to superimpose "semi-text" processing in 20 ms steps. As shown in Figure 3:

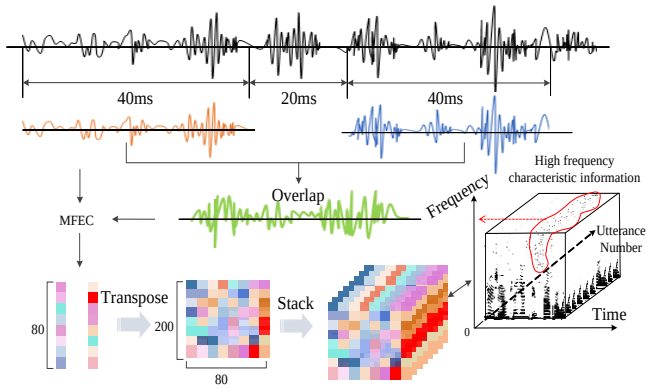

Fig. 3. Semi-textualization flow chart

From 8-second sound samples, 200 temporal feature sets (each feature set forming 80 MFEC features) can be obtained to form an input speech feature map. The dimension of each input feature map is $80 \times 200 \times \zeta$, which is composed of 200 input frames and their corresponding spectral features, where ζ is the number of utterances used to simulate speakers during the development and registration phases and represents the number of voices spoken by each person. Here, we set ζ to 40. In Figure 3, it can be seen that the semi-textualized three-dimensional speaker characteristics are preserved and enhanced at high frequencies.

C. *Design 3DCNN model*

Convolutional neural networks have a good ability to recognize biological signals [31]. They generally contain two important components: a convolutional layer and a pooling layer. Through convolution calculation, the local information in the speaker's original data is extracted and enhanced [32], and then the pooling layer compresses the enhanced features to reduce the dimension to reduce the number of network computations [33].

In this paper, instead of using traditional speaker recognition to process speech signals, the speech signals are used as the input to the convolutional neural network to perform one-dimensional convolution calculations, but the time-frequency-discourse volume three-dimensional data are used as the model input to introduce three-dimensional convolution. The kernel thus designs a new 3DCNN model structure to fully extract the speaker's speech features. This new convolution core is not a traditional N*N structure, but a new structure designed according to the speaker's speech characteristics. Its length and width are not equal. The 3DCNN network parameter structure designed in this paper is shown in Figure 4.

In Figure 4, the number of convolution kernels in the first two layers is set to 16, and their size is 3x1x5 and 3x9x5 three-dimensional convolution kernels, respectively, which can perform three-dimensional convolution on the time-frequency-discourse volume of the speaker's speech signal. To extract the deep-level features of the speaker, the number of the third and fourth convolution kernels is set to 32, their sizes are 3x1x4 and 3x8x1, and the pooling process is performed every two layers. In addition, the steps of the first four layers are alternately 1x1x1 and 1x2x1, which can not only fully extract the personality characteristics of the speaker but also ensure parameter learning efficiency. Additionally, each layer of the network is also provided with a BN layer to normalize the data to ensure that the stability of the parameters is avoided to avoid the problem of gradient disappearance or explosion.

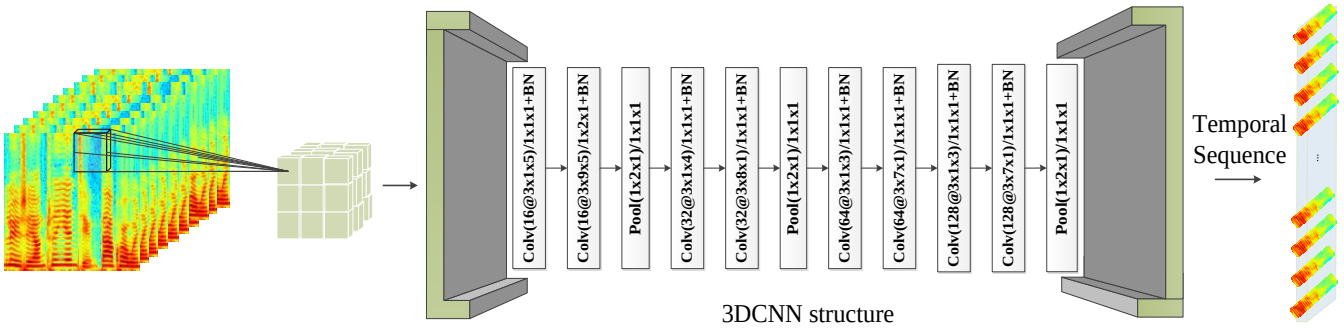

Fig. 4. 3DCNN model structure diagram

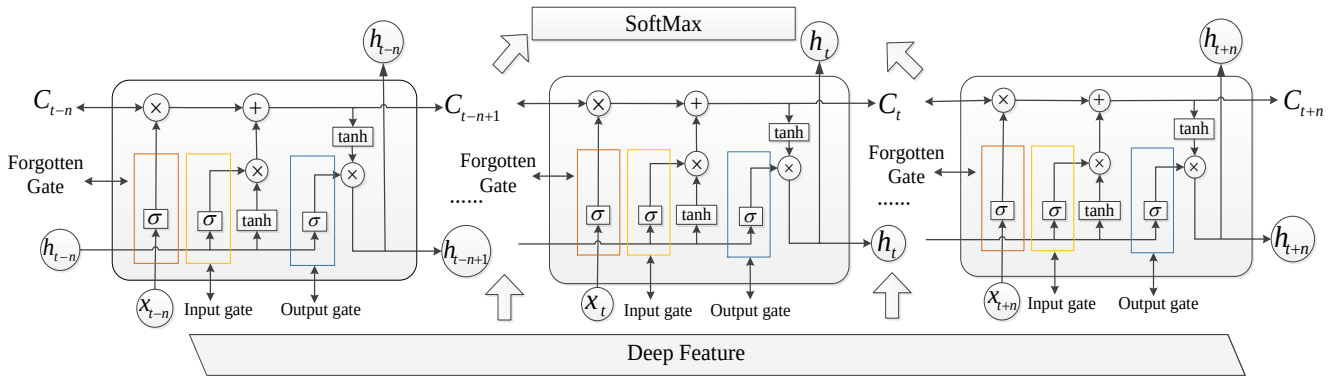

Fig. 5. LSTM network structure diagram

The number of convolution kernels of the fifth and sixth layers is set to 64, and their sizes are 3x1x3 and 3x7x1, respectively; the step size is set to 1x1x1; and the number of convolution kernels of the seventh and eighth layers is set to 128. Its size and step size are the same as those of the previous two layers, and each layer of the network also has a BN layer, which is finally pooled to obtain the speaker's deep personality characteristics.

D. Design LSTM

The data after convolution pooling is a sequence form, so it needs to be further processed before it can be recognized. The recurrent neural network RNN can process the sequence information well [34]; however, [35] showed that the RNN's ability to learn the context of the speech signal is weak and the gradient disappears easily during the training phase [36]. To solve this problem, a LSTM network was introduced for time-series feature extraction to prevent over-learning the network. Dropout was introduced inside the circulating neural unit to disconnect the connections made within the neuron by 10% to prevent the gradient from disappearing. The last layer to access the network is the softmax fully connected layer, which recognizes the speaker's identity. Its LSTM structure is shown in Figure 5.

Where i_t represents the input gate, f_t represents the forget gate, the old speaker features are discarded in the speaker model, and C_t represents the neuron activation representing the memory unit of this layer. C' represents the input speaker information, o_t represents the output gate, and h_t represents the speaker information output by the current network. σ is the activation function sigmoid, and W is the weight matrix of each stage. The LSTM network can perform context learning on deep feature sequences and improve the recognition rate [36]. The specific formulas of each part are as follows.

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}C_{t-1}) \quad (1)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}C_{t-1}) \quad (2)$$

$$C_t = f_t \odot C_{t-1} + C' \quad (3)$$

$$C' = i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1}) \quad (4)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}C_t) \quad (5)$$

$$h_t = o_t \odot \tanh(C_t) \quad (6)$$

$$m_t = \tanh(C_t) \quad (7)$$

LSTM has strong forward learning ability not only its forward propagation algorithm learning ability, but also its backward propagation learning ability. To more intuitively demonstrate the back propagation algorithm of LSTM, Figure 5 can be transformed into Figure 6.

For known $\partial J/\partial y_t$, $\partial J/\partial c_{t+1}$, $\partial J/\partial o_{t+1}$, $\partial J/\partial f_{t+1}$ and $\partial J/\partial \tilde{i}_{t+1}$ when calculating the gradient of a node, you should first find the output node of that node, then calculate the gradient of all output nodes multiplied by the gradient of the output node to that node, and finally add up to get the gradient of that node. For example, when calculating $\partial J/\partial h_t$, find all the output nodes of a node h_t , then calculate the gradient (y_t , o_{t+1} , f_{t+1} , $\tilde{i}_{t+1}$) of the output nodes separately, then calculate the product of the gradient ($\partial J/\partial h_t$) of the output node and the gradient corresponding to the output node separately (for example $(\partial J/\partial y_t)W_{yh}^T$), and add them together

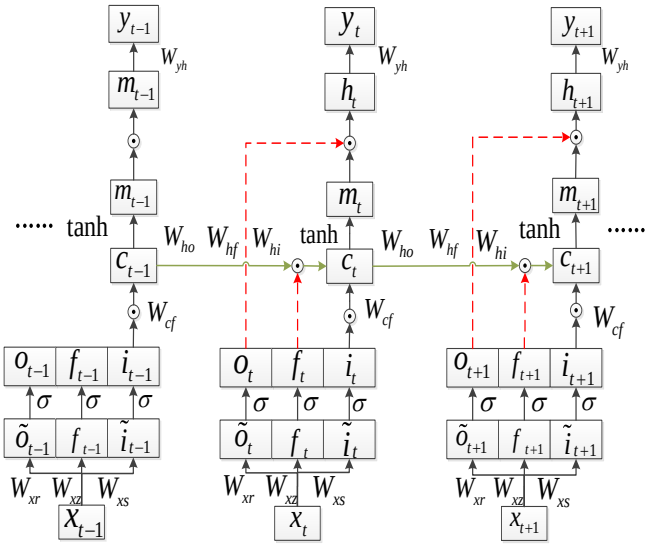

Fig. 6. LSTM network signal flow diagram

to get the gradient of the nodes:

$$\frac{\partial J}{\partial h_t} = \frac{\partial J}{\partial y_t}W_{yh}^T + \frac{\partial J}{\partial o_{t+1}}W_{ho}^T + \frac{\partial J}{\partial f_{t+1}}W_{hf}^T + \frac{\partial J}{\partial \tilde{i}_{t+1}}W_{hi}^T \quad (8)$$

Similarly, the gradient of other nodes at t-Time can be obtained:

$$\begin{cases} \dfrac{\partial J}{\partial h_t} = \dfrac{\partial J}{\partial y_t}W_{yh}^T + \dfrac{\partial J}{\partial o_{t+1}}W_{ho}^T + \dfrac{\partial J}{\partial f_{t+1}}W_{hf}^T + \dfrac{\partial J}{\partial \tilde{i}_{t+1}}W_{hi}^T & (9) \\ \dfrac{\partial J}{\partial m_t} = \dfrac{\partial J}{\partial h_t} \odot o_t & (10) \end{cases}$$

$$\begin{cases} \frac{\partial J}{\partial c_t} = \frac{\partial J}{\partial m_t}\frac{dm_t}{dc_t} + \frac{\partial J}{\partial \tilde{i}_{t+1}} \odot f_{t+1} + \frac{\partial J}{\partial f_{t+1}} W_{cf}^T + \frac{\partial J}{\partial \tilde{i}_{t+1}} W_{ci}^T & (11) \\ \frac{\partial J}{\partial f_t} = \frac{\partial J}{\partial c_t} \odot c_{t-1} \\ \frac{\partial J}{\partial o_t} = \frac{\partial J}{\partial h_t} \odot m_t \end{cases} \rightarrow \begin{cases} \frac{\partial J}{\partial f_t} = \frac{\partial J}{\partial f_t}(1-f_t) & (12) \\ \frac{\partial J}{\partial o_t} = \frac{\partial J}{\partial o_t}(1-o_t) & (13) \end{cases}$$

$$\frac{\partial J}{\partial x_t} = W_{ho}^T \frac{\partial J}{\partial \tilde{o}_t} + W_{hf}^T \frac{\partial J}{\partial \tilde{f}_t} + W_{hi}^T \frac{\partial J}{\partial \tilde{i}_t} \quad (14)$$

Gradients for parameters:

$$\begin{cases} \frac{\partial J}{\partial W_{ho}} = h_t^T \frac{\partial J}{\partial o_{t+1}} \\ \frac{\partial J}{\partial W_{hf}} = h_t^T \frac{\partial J}{\partial f_{t+1}} \\ \frac{\partial J}{\partial W_{hi}} = h_t^T \frac{\partial J}{\partial \tilde{i}_{t+1}} \end{cases} \begin{cases} \frac{\partial J}{\partial W_{yh}} = h_t^T \frac{\partial J}{\partial y_t} \\ \frac{\partial J}{\partial W_{cf}} = c_t^T \frac{\partial J}{\partial f_{t+1}} \\ \frac{\partial J}{\partial W_{ci}} = c_t^T \frac{\partial J}{\partial \tilde{i}_{t+1}} \end{cases} \begin{cases} \frac{\partial J}{\partial W_{xo}} = x_t^T \frac{\partial J}{\partial o_t} & (15) \\ \frac{\partial J}{\partial W_{xf}} = x_t^T \frac{\partial J}{\partial f_t} & (16) \\ \frac{\partial J}{\partial W_{xi}} = x_t^T \frac{\partial J}{\partial \tilde{i}_t} & (17) \end{cases}$$

III. Results and Results Analysis

A. Experimental data

The experimental data used in this article are based on the open-source voice dataset AISHELL-1, which was jointly recorded by speakers from different regions of China under the real environment. The recording devices used are a high-fidelity microphone (44.1 kHz, 16 bit) Android system, mobile phone (16 Hz, 16 bit), and iOS system mobile phone (16 kHz, 16 bit). This article uses the voice information of 20 different speakers for a total of 15 hours, including 12 hours in the training phase and 3 hours in the test phase, with a sampling rate of 16 kHz.

This article does not use the speaker's voice directly as the input to the network model but treats each speaker's voice information as "semi-text" processing, which is conducive to subsequent contextual learning of the LSTM network and thus improves the speaker's recognition rate. In this paper, the time of 40 ms is used, and the 8 s speaker speech is superimposed at 20 ms intervals, which is counted as 1 sample. In this experiment, there were 10 male speakers and 10 female speakers, with a total of 13,564 samples.

B. Experimental evaluation index

The evaluation index used in this experiment is the speaker recognition rate $accuracy = n / N \times 100\%$, where N represents the total number of speakers and n represents the number of people who are recognized correctly.

C. Experimental platform construction

In this paper, the "semi-text" processing of the speaker's speech signal and three-dimensional data conversion are implemented on the Windows 10 system (64 bit) through MATLAB 2014b software. The training and testing of the deep learning neural network are in Ubantu16.04 (64 bit), the running memory is 16 GB, and the GPU device is 4 GXT080ti. In the model training and optimization stage, this paper uses an initial learning rate of 0.01, $\beta 1$ =0.9, $\beta 2$ =0.999, ε =10e-8, and the optimizer is Adam. The number of nodes in the fully connected layer is set to 3,026. To prevent the gradient from disappearing in the training process, this paper uses the dropout method to set the initial value to 0.95 and applies it to each layer of the network. The cross-entropy loss function is selected when calculating the loss function.

D. Experimental model parameters

To verify the effectiveness of the first mock exam model in comparison with the single model, three models of 3DCNN, LSTM and 3DCNN-BLSTM are designed and compared with the 3DCNN-LSTM network designed in this paper. The network structure and parameters of the four models are shown in TABLE Ⅰ.

TABLE Ⅰ

Model	Input signal data	Network structure	Specific structural parameters
Mode1	3D speech array	3DCNN	4*(Cov3D+BN)+Pool+4*(Cov3D+BN)+Pool+FC+Softmax
Mode2	Speech sequence	LSTM	4*LSTM+Softmax
Mode3 (ours)	3D speech array	3DCNN-LSTM	4*(Cov3D+BN)+Pool+4*(Cov3D+BN)+Pool+4*LSTM+FC+Softmax
Mode4	3D speech array	3DCNN-BLSTM	4*(Cov3D+BN)+Pool+4*(Cov3D+BN)+Pool+2*(4*LSTM)+FC+Softmax

E. Experimental comparison

To explore the performance advantages and disadvantages between different models, this paper first completes the training of the four models on the training set, then finds the optimal number of iterations, and finally compares the performance in the test set. The figure below shows the performance of each model in the training set. It is not difficult to see from Figure 7 that model 1 and model 2 achieve the best recognition effect at 380 times and 410 times, respectively, while model 4 and the model provided in this paper gradually tend to be stable after 560 and 520 training times, respectively, achieving the best effect. This is because compared with model 3 and the composite model structure proposed in this paper, the network structure of model 1 and model 2 is relatively simple and can reach the stable state earlier in the training process. Compared with the model provided in this article, model 4 introduces a two-way LSTM network structure. The parameter updates of the input gate, output gate, and forget gate need to be repeatedly updated to confirm whether the recognition effect is the best. Therefore, 3DCNN-BLSTM needs to learn more parameters, converges more slowly, and consumes the longest time.

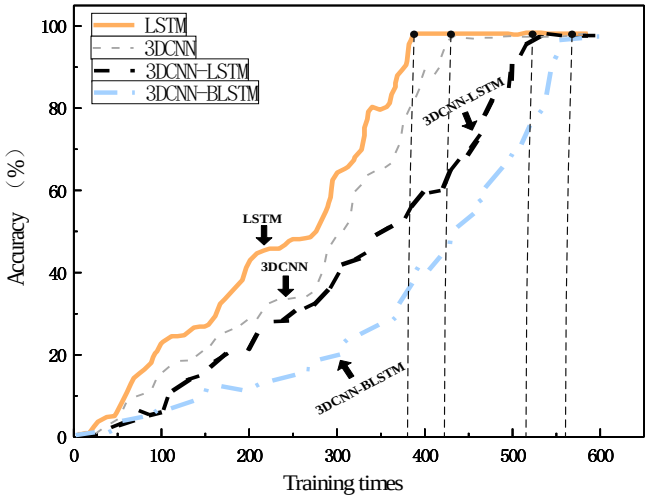

Fig. 7. Performance of the training

The second group of experiments is set up in this paper. Under the condition of the best training effect, the four groups of models are tested for the best recognition effect under 380 training times, 410 times for model 2, 520 times for model 3 and 560 times for model 4. The results obtained in the test set are shown in Figure 8.

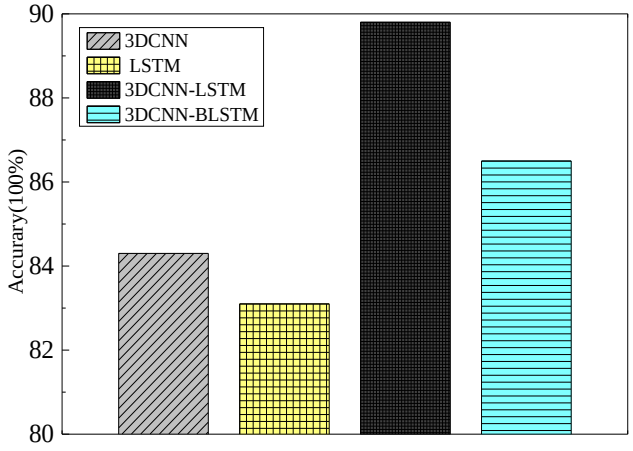

Fig. 8. Performance of different models on test set

It can be seen in Figure 9 that the performances of the combined models 3DCNN-LSTM and 3DCNN-BLSTM are better than that of the single models 3DCNN and LSTM. The performance of 3DCNN-LSTM improved by 9.11% and 8.06% compared to the two single models. Compared with the two single models, the BLSTM performance improved by 5.10% and 4.09%, which proves the effectiveness of the deep speaker recognition method in the combined mode.

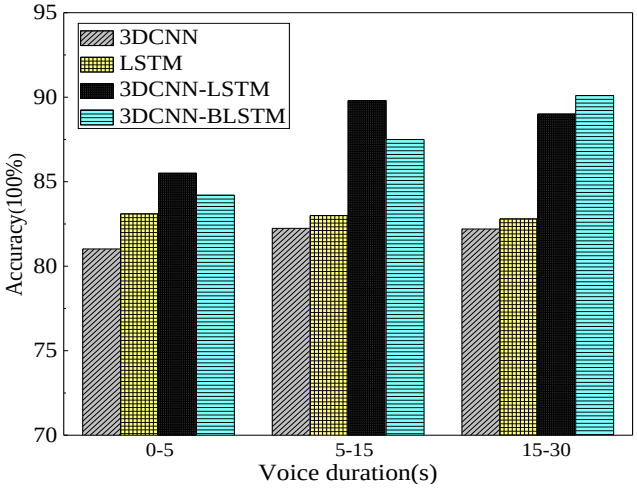

Fig. 9. Performance of the model under different durations

The third set of experiments in this paper tests speakers with different speech durations under the best performance because even with the same model, the speech duration of different speakers will have different effects. Therefore, three sets of speech durations of [0s-5s], [5s-15s] and [15s-30s] were set in this paper. The test was conducted 20 times, and the final results were averaged as shown in Figure 9.

It is not difficult to see from Figure 9 that the model proposed in this paper performs best in [5s-15s], while in [15s-30s], because of the increase in training time, the single-layer LSTM network structure cannot fully learn the upper and lower text information, while the double-layer structure of 3DCNN-BLSTM network structure shows better performance in long text environments, and the performance is 2.97% higher than the model proposed in this paper, which benefits from the powerful context learning ability of BLSTM network power.

F. Experimental comparison of different methods

The experimental comparison methods selected in this paper include not only GMM-UBM [38], [39] and i-vector+PLAD in traditional algorithms [40]-[42] but also popular deep learning algorithms [43]-[46], including TDNN, DNN, CNN and their fusion products based on embedding features[47]-[53]. Table Ⅱ shows the results of recognition rate of each method and the performance improvement value of our method compared with other methods (positive number corresponds to performance improvement, and complex number corresponds to performance reduction).

Because the parallel structure of the three-dimensional convolutional neural network and long short-term memory

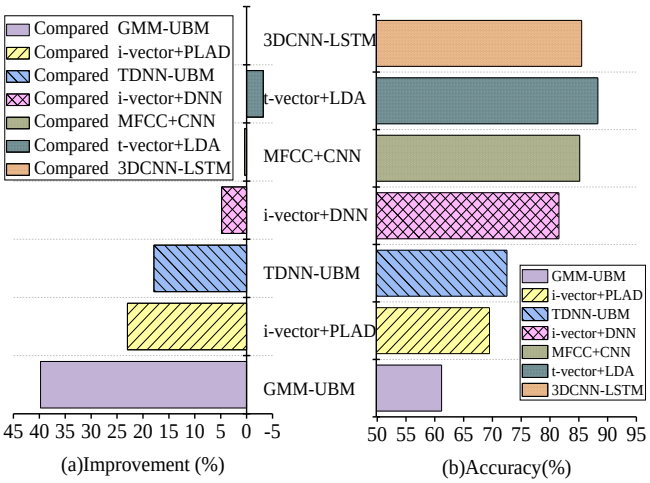

Fig. 10. 0s-5s speech duration

TABLE Ⅱ

Duration	METHOD						
	Traditional methods			Embedding feature method			Ours
	GMM-UBM	i-vector+PLAD	TDNN-UBM	i-vector+DNN	MFCC+CNN	t-vector+LDA	3DCNN-LSTM
0s-5s	61.17%	69.50%	72.51%	81.55%	85.16%	88.32%	85.49%
5s-15s	74.00%	86.10%	80.19%	88.80%	83.14%	84.80%	89.80%
15s-30s	88.00%	87.58%	85.53%	93.90%	91.62%	—	88.98%

Duration	IMPROVEMENT						
	Traditional methods			Embedding feature method			Evaluation
	GMM-UBM	i-vector+PLAD	TDNN-UBM	i-vector+DNN	MFCC+CNN	t-vector+LDA	3DCNN-LSTM
0s-5s	39.76%	23.01%	17.90%	4.83%	0.39%	-3.20%	Good
5s-15s	21.35%	4.30%	11.98%	1.13%	8.01%	5.90%	Advanced
15s-30s	1.11%	1.60%	4.03%	-5.24%	-2.88%	—	Acceptable

network is used in this paper, the performance of the speaker recognition system is improved. The performance of the system is effectively improved compared with the traditional methods and the current popular methods. With the increase of the speaker's speech duration, the system recognition rate remains at a high level. In the short speech environment of 0s-5s, our method has obvious advantages over traditional algorithms in Figure 10. Compared with embedding feature, embedding feature is slightly better than MFCC+CNN combination and weaker than t-vectors combination embedding feature.

In the 5s-15s standard speaker recognition speech length, Figure 11 shows that our model is better than the traditional algorithm and embedding feature method, but it is worth noting that the traditional i-vector + PLAD algorithm still has better performance in this speech duration range, and its performance is better than that of partial embedding feature method.

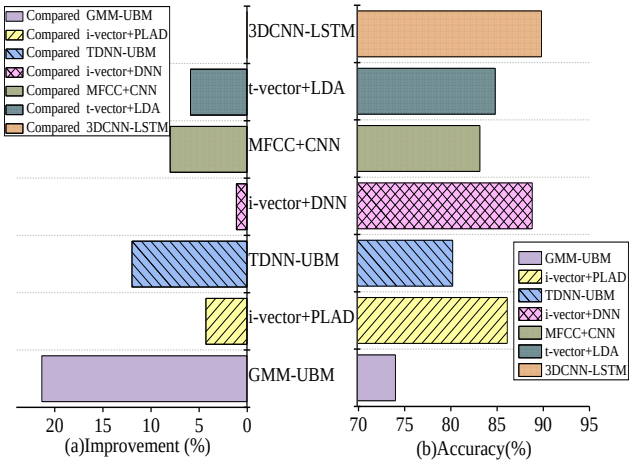

Fig. 11. 5s-15s speech duration

Figure 12 shows that our method is still ahead of the traditional speaker recognition method in the 15s-30s speech durations, but compared with embedding feature method, our performance has no advantage. The t-vector + LDA method has no reliable data, but because this method is a fusion product of i-vector+DNN and MFCC+CNN methods, it is reasonable to speculate that this method still has excellent performance in such a speech duration environment.

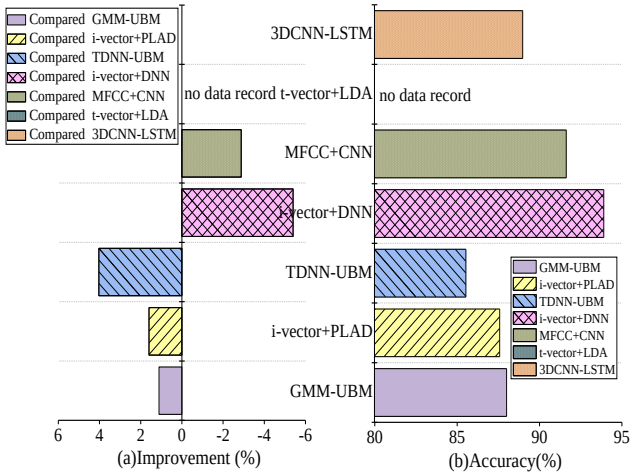

Fig. 12. 15s-30s speech duration

IV. Conclusion

This paper proposes a "semi-textualized" 3DCNN-LSTM network model. First of all, the input of the network is the time-frequency-speech volume data. Compared with the traditional i-vector based on MFCC transformation, the 3D speaker features can better preserve and enhance high frequency information. Secondly, the unique parameter structure of the 3DCNN convolution kernel is designed to ensure that the key information of the speaker can be extracted. Moreover, compared with the single network model structure, the proposed model(3DCNN-LSTM) can effectively improve the speaker recognition rate. Last but not least, compared with the traditional speaker recognition network model and embedding features, the model has obvious performance advantages in short duration and middle duration of speaker recognition. Although the performance of long duration speaker recognition is not as good as embedding features, it is still superior to traditional speaker recognition methods. For the increase of speaker speech durations, the system performance remains at a high level and robustness is stronger, which is of great significance in practical application. However, we still need to address the following points:

(1) In the long speech environment of this model, the inadequate ability of the back-end LSTM network to learn contextual text leads to a decrease in recognition accuracy. How to optimize the back-end is the next research direction.

(2) Noise elimination is also an important task in speaker recognition, which determines the final recognition effect. The optimization of the noise reduction algorithm in the preprocessing part is also the next research direction.

References

[1] Todkar S P , Babar S S , Ambike R U , et al. Speaker Recognition Techniques: A Review[C]// International Conference for Convergence in Technology. 2018.

[2] Chauhan N , Isshiki T , Li D . Speaker Recognition Using LPC, MFCC, ZCR Features with ANN and SVM Classifier for Large Input Database[C]// 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS). IEEE, 2019.

[3] Leu F Y , Lin G L . An MFCC-Based Speaker Identification System[C]// IEEE International Conference on Advanced Information Networking & Applications. IEEE, 2017.

[4] Tagashira, Mizuho, and Takafumi Nakagawa. "Biometric Authentication Based on Auscultated Heart Sounds in Healthcare." IAENG International Journal of Computer Science, vol.47, no.3 .pp.343-349, 2020.

[5] Rehman F U , Kumar C , Kumar S , et al. VQ based comparative analysis of MFCC and BFCC speaker recognition system[C]// International Conference on Information & Communication Technologies. 2017.

[6] Jokinen E., Saeidi R., Kinnunen T., et al. Vocal effort compensation for MFCC feature extraction in a shouted versus normal speaker recognition task[J]. Computer Speech & Language, 2019, 53(1): 1-11.

[7] Li Q., Yang Y., Lan T., et al. MSP-MFCC: energy-efficient MFCC feature extraction method with mixed-signal processing architecture for wearable speech recognition applications[J]. IEEE Access, 2020, 8(3): 48720-48730.

[8] Dhanush B. K., Suparna S., Aarthy R., et al. Factor analysis methods for joint speaker verification and spoof detection[C]//In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE,2017: 5385-5389.

[9] Vestman V., Kinnunen T., Hautamäki R. G., et al. Voice mimicry attacks assisted by automatic speaker verification[J]. Computer Speech & Language, 2020, 59:36-54.

[10] Chakroun R , Zouari L B , Frikha M , et al. Improving text-independent speaker recognition with GMM[C]// International Conference on Advanced Technologies for Signal & Image Processing. IEEE, 2016.
[11] Zhang Xingyu, Zou Xia, Sun Meng, et all. Noise robust speaker recognition based on adaptive frame weighting in GMM for i-vector extraction[J]. IEEE Access, 2019, 7(2): 27874-27882.
[12] Gupta M., Bharti S. S., Agarwal S. Gender-based speaker recognition from speech signals using GMM model[J]. Modern Physics Letters B, 2019, 33(35):1950438.
[13] Ramoji S., Ganapathy S. Supervised i-vector modeling for language and accent recognition[J]. Computer Speech & Language, 2020, 60(3): 101030.
[14] Ghahabi O., Hernando J. Deep learning backend for single and multisession i-vector speaker recognition[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2017, 25(4): 807-817.
[15] Yao S , Zhou R , Zhang P , et al. Discriminatively learned network for i-vector based speaker recognition[J]. Electronics Letters, 2018, 54(22):1302-1304.
[16] Liu Zheli, Wu Zhendong, Li Tong, et al. GMM and CNN hybrid method for short utterance speaker recognition[J]. IEEE Transactions on Industrial informatics, 2018, 14(7): 3244-3252.
[17] Chen, Rung-Ching. "Using Deep Learning to Predict User Rating on Imbalance Classification Data." IAENG International Journal of Computer Science, vol.46, no.1, pp.109-117, 2019.
[18] Sasikala, V. , and V. L. Prabha . "Bee swarm based feature selection for fake and real fingerprint classification using neural network classifiers." IAENG International Journal of Computer Science, vol.42, no.4, pp.389-403, 2015.
[19] Guan, Huanxin, et al. "Multiple Faults Diagnosis of Distribution Network Lines Based on Convolution Neural Network with Fuzzy Optimization." IAENG International Journal of Computer Science, vol.47, no.3, pp.567-571, 2020.
[20] Novotny O , Plchot O , Glembek O , et al. Analysis of DNN Speech Signal Enhancement for Robust Speaker Recognition[J]. Computer speech and language, 2019, 58(NOV.):403-421.
[21] Identity Vector Extraction Using Shared Mixture of PLDA for Short-Time Speaker Recognition[J]. Chinese Journal of Electronics, 2019, 28(02):138-144.
[22] Kanagasundaram A , Sridharan S , Ganapathy S , et al. A Study on Pairwise LDA for X-vector based Speaker Recognition[J]. Electronics Letters, 2019.
[23] Snyder D., Garcia-Romero D., Sell G., et al. Speaker recognition for multi-speaker conversations using x-vectors[C]//In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE,2019: 5796-5800.
[24] Garcia-Romero D , Snyder D , Sell G , et al. x-Vector DNN Refinement with Full-Length Recordings for Speaker Recognition[C]// Interspeech 2019. 2019.
[25] Xue S , Abdel-Hamid O , Jiang H , et al. Direct adaptation of hybrid DNN/HMM model for fast speaker adaptation in LVCSR based on speaker code[C]// 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014.
[26] Snyder D , Garcia-Romero D , Povey D . Time delay deep neural network-based universal background models for speaker recognition[C]// 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). IEEE, 2015.
[27] Kanagasundaram A , Sridharan S , Ganapathy S , et al. A Study on Pairwise LDA for X-vector based Speaker Recognition[J]. Electronics Letters, 2019.
[28] Zaman, Lukman, et al. "Modeling Basic Movements of Indonesian Traditional Dance Using Generative Long Short-Term Memory Network." IAENG International Journal of Computer Science, vol.47, no.2, pp.262-270, 2020.
[29] Cumani S , Laface P . Speaker Recognition Using e–Vectors[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2019.
[30] Torfi, Amirsina , J. Dawson , and N. M. Nasrabadi . "Text-Independent Speaker Verification Using 3D Convolutional Neural Networks." (2017).
[31] An N N , Thanh N Q , Liu Y . Deep CNNs with Self-Attention for Speaker Identification[J]. IEEE Access, 2019, PP(99):1-1.
[32] Lukic Y , Vogt C , Durr O , et al. Speaker identification and clustering using convolutional neural networks[C]// IEEE International Workshop on Machine Learning for Signal Processing. IEEE, 2016.
[33] Huangkang C , Ying C . Speaker Recognition based on Multimodal LSTM using Depth-Gate[C]// IEEE Advanced Information Technology, Electronic & Automation Control Conference. IEEE, 2018.
[34] Chen Huangkang, Chen Ying. Speaker Identification Based on Multimodal Long Short-Term Memory with Depth-Gate[J]. Laser & Optoelectronics Progress, 2019, 56(3):031007.
[35] Hashida, Shuichi, Keiichi Tamura, Tatsuhiro Sakai. “Classifying Tweets using Convolutional Neural Networks with Multi-Channel Distributed Representation,” IAENG International Journal of Computer Science, vol.46, no. 1, pp.68-75, 2019
[36] Miao X , Mcloughlin I . LSTM-TDNN with convolutional front-end for Dialect Identification in the 2019 Multi-Genre Broadcast Challenge[J]. 2019.
[37] Dovydaitis L , Vytautas Rudžionis. Building LSTM neural network based speaker identification system[J]. 2018.
[38] Akula A , Apsingekar V R , Leon P L D . Speaker Identification in Room Reverberation Using GMM-UBM[C]// Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop, 2009. DSP/SPE 2009. IEEE 13th. IEEE, 2009.
[39] Li, Rongjin, et al. "Improving the Generalized Performance of Deep Embedding for Text-Independent Speaker Verification." 2018 12th IEEE International Conference on Anti-counterfeiting, Security, and Identification (ASID). IEEE, 2018.
[40] Hanilci C , Celiktas H . [IEEE 2018 26th Signal Processing and Communications Applications Conference (SIU) - Izmir, Turkey (2018.5.2-2018.5.5)] 2018 26th Signal Processing and Communications Applications Conference (SIU) - Turkish text-dependent speaker verification using i-vector/PLDA approach[C]// 2018:1-4.
[41] Chakroun R , Frikha M . Efficient text-independent speaker recognition with short utterances in both clean and uncontrolled environments[J]. Multimedia Tools and Applications, 2020(5).
[42] Poddar, Arnab, Md Sahidullah, and Goutam Saha. "Performance comparison of speaker recognition systems in presence of duration variability." 2015 Annual IEEE India Conference (INDICON). IEEE, 2015.
[43] Liu H , Zhao L . A Speaker Verification Method Based on TDNN–LSTMP[J]. Circuits Systems & Signal Processing, 2019.
[44] Jahangir R , Teh Y W , Memon N A , et al. Text-independent Speaker Identification through Feature Fusion and Deep Neural Network[J]. IEEE Access, 2020, PP(99):1-1.
[45] Novoselov S , Kudashev O , Schemelinin V , et al. Deep CNN based feature extractor for text-prompted speaker recognition[J]. 2018.
[46] Jagiasi, Rohan, et al. CNN based speaker recognition in language and text-independent small scale system[C]// 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC). IEEE, 2019.
[47] Toruk, Muhammet Mesut, and Ramazan Gokay. "Short Utterance Speaker Recognition Using Time-Delay Neural Network." 2019 16th International Multi-Conference on Systems, Signals & Devices (SSD). IEEE, 2019.
[48] Chen X , Wang Y , Wang L , et al. Speaker Recognition Method Based on Statistical Features of Spectrograms and CNN[C]// the 3rd International Conference. 2019.
[49] Guo J , Xu N , Qian K , et al. Deep neural network based i-vector mapping for speaker verification using short utterances[J]. Speech Communication, 2018.
[50] Kumari, TR Jayanthi, and H. S. Jayanna. "Comparison of LPCC and MFCC features and GMM and GMM-UBM modeling for limited data speaker verification." 2014 IEEE International Conference on Computational Intelligence and Computing Research. IEEE, 2014.
[51] Wang, Wenchao, et al. "Multiple Temporal Scales Based Speaker Embeddings Learning for Text-dependent Speaker Recognition." ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019.
[52] Zhang, Chunlei, et al. "UTD-CRSS systems for 2018 NIST speaker recognition evaluation." ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019.
[53] Wu Zhendong, Pan Shucheng, and Zhang Jianwu. "Continuous speech voiceprint recognition based on CNN." Telecommunications Science 33.003 (2017): 59-66.