GMM and CNN Hybrid Method for Short Utterance Speaker Recognition

Zheli Liu, Zhendong Wu, Tong Li*, Jin Li, Chao Shen

***Abstract*—Recently, the speaker recognition technique has been widely attractive for its extensive application in many fields, such as speech communications, domestics services, and smart terminals. As a critical method, the Gaussian Mixture Model (GMM) makes it possible to achieve the recognition capability that is close to the hearing ability of human in a long speech. However, the GMM is failing to recognize a short utterance speaker with a high accuracy. Aiming at solving this problem, in this paper, we propose a novel model to enhance the recognition accuracy of the short utterance speaker recognition system. Different from traditional models based on the GMM, we design a method to train a Convolutional Neural Network (CNN) to process spectrograms, which can describe speakers better. Thus, the recognition system gains the considerable accuracy as well as the reasonable convergence speed. The experiment results show that our model can help to decrease the equal error rate of the recognition from 4.9% to 2.5%.**

***Index Terms*—speaker verification, UBM-MAP-GMM, spectrogram, CNN.**

I. Introduction

VOICEPRINT verification, one of the most popular biometric technologies [1], [2], has been widely used in many areas [2], [3], such as the speech recognition [4]. One of its primary objectives is to determine whether the test voice from the speaker is allowable. Generally speaking, speaker verification and speaker identification are two important branches of speech recognition [5]. The technologies of speaker verification have been studied extensively in recent years.

However, it is not easy to collect an appropriate speech in some application scenarios. The current speaker verification system can achieve a fairly good recognition rate only when the testing utterances are long enough. In a short language environment [6], the recognition rate drops obviously. In fact, a short utterance means the utterance contains inadequate acoustic features. In such a situation, the traditional speaker models based on statistical properties fail to perfectly describe speakers. Although the traditional voiceprint model has the obvious characteristic specificity, it is still easily disturbed by the noise because the number of features is too few.

Zheli Liu is with College of Computer and Control Engineering, Nankai University, China. E-mail: liuzheli@nankai.edu.cn.

Zhendong Wu is with School of Cyberspace, Hangzhou Dianzi University, China. E-mail: wzd@hdu.edu.cn.

Tong Li and Jin Li are with School of Computer Science and Educational Software, Guangzhou University, China. E-mail: litongziyi@mail.nankai.edu.cn (corresponding author), jinli71@gmail.com.

Chao Shen is with School of Electronic and Information Engineering, Xi'an JiaoTong University, China. E-mail: cshen@sei.xjtu.edu.cn.

Manuscript received Nov. 10, 2017

Intuitively, the deep learning model is helpful for solving this problem due to its deep feature learning ability. In more details, it can achieve the strong anti-interference goal by excavating a larger number of voiceprint features. However, the training process requires a large number of samples and the characteristic specificity is not obvious, which results the deep learning cannot work directly. Considering the instability of the acquisition of short utterance features and the possible shortage of training samples, we focus on how to design the voiceprint model and deep-learning model for the short utterance recognition, which can effectively overcome the shortcomings of less training samples and susceptible to interference.

A. Related Works

Great progress has been made in speaker recognition and some speaker models have been proposed. The Gaussian Mixture Model (GMM), GMM-Universal background model (GMM-UBM) and SVM can accurately describe the target speaker. Recently, Joint Factor Analysis (JFA) [7] and *i*-vector models [8] have also been proposed. Now, researchers are starting to apply speaker recognition and other biometric identification technologies to secure authorization [9], data privacy [10], cloud computing security [11], and video security [12].

However, the speaker models mentioned above all have some limitations in the short language environment. For example, the GMM speaker recognition technology, which uses the segmented statistical features of the speech spectrum to identify the speaker, is difficult to obtain good results in short utterance. That is, the GMM model has the limitation of the volatility of the short-term speech in the spectral statistics. With the rise of deep learning technology, deep learning of voiceprint features using deep neural networks may solve the speaker recognition problem for short utterance. Even so, to the best of our knowledge, there is lack of effective recognition method for the short language environments.

The Convolutional Neural Network (CNN) is a kind of multi-layer, shared weight, task-oriented learning depth neural network. Compared to shallow network, the CNN has its unique advantages in characterizing complex functions, which can help us to complete complex, high-level abstract artificial intelligence tasks [13]–[20]. Convolution neural networks have yielded remarkable results in many pattern recognition problems, such as audit classification [21], finger vein recognition [22], action recognition [23], object recognition [24] and traffic sign classification [25]. A convolution neural network can

complete the feature extraction task while completing the classification task [26].

In general, there are two categories of features for speech recognition, which are Linear Prediction Cepstrum Coefficients (LPCC) and Mel Frequency Cepstrum Coefficient (MFCC). However, when they are directly applied to the short speech environment, they meet performance bottlenecks. To our best knowledge, there is lack of effective short utterance speaker verification model. Some researchers paid much more attention to rhythmic features to improve recognition accuracy for the short language environment. Based on massive experiments, they found that speech spectrogram contains useful and discriminative information. In another word, the knowledge of speech spectrogram has a great importance for distinguishing different people. Unfortunately, there is still no method to train a CNN for obtaining a specific speaker model from the spectrograms.

B. Contribution

To fill the gap between the acquisition of short utterance features and the possible shortage of training samples, we propose a novel GMM-CNN hybrid method for short utterance speaker recognition. The proposed method can increase the recognition accuracy of the short utterance speaker recognition system.

Our contributions can be concluded as follows:

- We propose an initial alignment method for short utterance feature, which can improve the effect of short utterance recognition.
- We design a novel method to transform a speech spectrogram to a fixed-size image, so that a CNN can work on the spectrogram.
- Based on the CNN, we design a short utterance recognition method to make better use of speech feature information and to obtain high speech recognition accuracy with a few training samples.

II. PRELIMINARY RESEARCH

A. Speaker Recognition

The process of speaker recognition consists of three stages that are the acoustic feature extraction, the statistical modeling, and the fractional calculation. The process is shown in Fig.1.

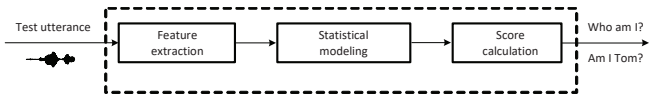

Figure 1. Speaker Recognition Process

1) Acoustic feature extraction: The aim of this stage is to extract Mel-Frequency Cepstral Coefficients (MFCC).

The length of voice frame usually decided by its duration. If shifting the frame costs 10 ms and the voice lasts not more than 25 ms, this voice is considered to be a short time stationary.

The pre-emphasis part can be seen as a high pass filter which is equivalent to

$$H(z) = 1 - az^{-1} \tag{1}$$

where a is a pre-emphasis coefficient (usually in the interval [0.95,0.97]) and the frequency warping can describe the natural sound.

The hamming window $\omega\cdot$ shown as follows is used to smooth edge of framed signals.

$$\omega(k) = \left[0.54 - 0.46\cos\left(\frac{2\pi k}{M-1}\right)\right] R_M(k) \tag{2}$$

In voice processing, the Mel-Frequency Cepstrum (MFC) has effect on the short-term power spectrum of speech, which is on the basis of a cosine transform on a nonlinear mel-scale of frequency. The MFC can be obtained as follows.

$$F_{mel}(f) = 2595 \cdot \log(1 + \frac{f}{700}) \tag{3}$$

Its coefficients are derived from a type of cepstral representation of the speech clip, which is a nonlinear "spectrum-of-a-spectrum". The difference between Mel-frequency Cepstrum and cepstrum is whether the frequency band varies linearly. That of the MFC is closer to the human auditory system response to the sound than that of the cepstrum. The bands can be obtained as follows.

$$H_m(k) = \begin{cases} 0 & k < f(m-1) \\ \frac{k - f(m-1)}{f(m) - f(m-1)} & f(m-1) < k < f(m) \\ \frac{f(m+1) - k}{f(m+1) - f(m)} & f(m) < k < f(m+1) \\ 0 & k > f(m+1) \end{cases} \tag{4}$$

Note that M is the total number of triangular filters whose range is $0 \leq m < M$ in Eq. (4).

The function $f(\cdot)$ is the center frequency of a Mel band pass filter bank, and the value of the m-th bank is:

$$f(m) = \left(\frac{N}{f_s}\right) F_{mel}^{-1}\left(F_{mel}(f_l) + m\frac{F_{mel}(f_h) - F_{mel}(f_l)}{M+1}\right) \tag{5}$$

Note that N is the length of FFT, and f_h and f_l are the maximum frequency and the minimum frequency respectively.

The function F_{mel} has inverse function $F_{mel}^{-1}(b) = 700(e^{b/1125} - 1)$, which is the translation from Mel frequency to Hz frequency.

Overall, the MFCC is processed as shown in Fig. 2.

2) Statistical modeling: Given a series of feature vectors $X = \{x_1, ..., x_t, ..., x_m\}$, and speaker dependent model $\lambda = \{w_1, \mu_i, \sum_i\}$, the aim of this stage is to compute necessary statistical information. The coefficients in λ consist of iterative formulas for *weight*, *mean*, and *variance* respectively. Their values are shown as follows.

$$\omega_i = \frac{1}{m}\sum_{t=1}^{m} \Pr(i|x_t, \lambda) \tag{6}$$

$$\mu_i = \frac{\sum_{t=1}^{m} \Pr(i|x_t, \lambda) x_t}{\sum_{t=1}^{m} \Pr(i|x_t, \lambda)} \tag{7}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2799928, IEEE Transactions on Industrial Informatics

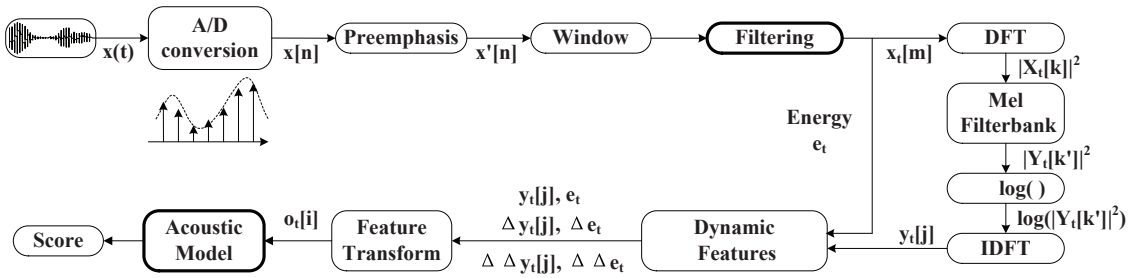

Figure 2. Extracting of Mel-Frequency Cepstral Coefficient

$$\Sigma_i = \frac{\sum_{t=1}^{m} \Pr(i|x_t,\lambda)(x_t-\mu_i)(x_t-\mu_i)'}{\sum_{t=1}^{m} \Pr(i|x_t,\lambda)} \tag{8}$$

Then, the posterior probability of the ith component is:

$$\Pr(i|x_t,\lambda) = \frac{p(x_t|i;\mu,\Sigma)p(i;\omega)}{\sum_{l=1}^{k} p(x_t|l;\mu,\Sigma)p(l;\omega)} \tag{9}$$

$$\Pr(i|x_t,\lambda) = \frac{\omega_i b_i(x_t)}{\sum_{l=1}^{k} \omega_l b_l(x_t)} \tag{10}$$

The score for making a decision is defined as follows.

$$\lg(L) = \lg(p(X|\lambda)) - \lg(p(X|\lambda_u)) \tag{11}$$

3) GMM-UBM: The Gaussian Mixture Model and Universal Background Model (GMM-UBM) algorithm is used to calculate scores, where 24-dimension MFCCs are used as acoustic characteristics. A UBM is a general model that needs to be trained first, so that parameters of the GMM (i.e., the speaker related model) are effectively estimated from the UBM via Expectation Maximization (EM) algorithm [27].

This GMM-UBM algorithm increases the recognition accuracy of the estimated model through iteratively changing the GMM parameters. The process of updating the parameter λ^* will not stop until it converges.

Let $O(\lambda,\lambda^*)$ denotes Jensen inequality, the parameters estimation is equivalent to maximize the function:

$$Q(\lambda,\lambda^*) = \sum_{k=1}^{m}\sum_{i=1}^{M} \frac{\omega_i p(x^k|\lambda,i)}{p(x^k|\lambda)} \cdot \left[\log \omega_i^* + \log p(x^k|\lambda^*,i)\right] \tag{12}$$

where $\omega_i p\left(x^k|\lambda,i\right) = p\left(x^k,i|\lambda\right)$. Let $\partial Q(\lambda,\lambda^*)/\partial\omega_i{}^* = 0$, the estimation is similar methods for estimating weights and covariances. In detail,

$$b_i(x^k) = \frac{\omega_i}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp(-\frac{1}{2}(x^k-\mu_i)'\Sigma_i^{-1}(x^k-\mu_i)) \tag{14}$$

$$\frac{\partial Q(\lambda,\lambda^*)}{\partial\theta} = \nabla_\theta \sum_{k=1}^{m}\sum_{i=1}^{D} \phi_i^k \cdot \log\frac{b_i(x^k)}{\phi_i^k} \tag{15}$$

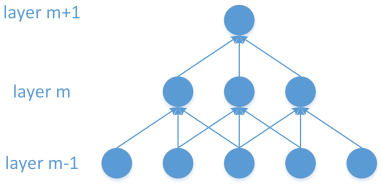

Figure 3. Sparse Connection Schematic

where $\theta \in \{\omega_i, \mu_i, \sum_i\}$ is learning parameters. Especially, λ is the Gaussian component, and the k is the number of features. Note that the speaker model is initialized randomly and each posteriori probability ϕ_i is iteratively computed by sample data primordially.

B. Convolution Neural Network

A Convolution Neural Network (CNN) can be adopted for extracting local features and pooling outputs by simulating the connection between visual cortex and unit hierarchy. Each layer in the CNN outputs convolutions or sub-samplings for stacking the features map of itself. The map of the current layer is detected by the previous layer and processed for the following layer. The CNN achieves a certain degree of transforming and deformation invariance due to its architecture for sharing weights and spatial sub-samplings. The local architecture of a CNN for the deep learning is shown in Fig.3 and Fig.4.

1) Sparse Connectivity: Since a CNN enhances the local connectivity between adjacent neurons, building a CNN can help to extract locally correlation features in space. That is, the convolution values of the local continuous cell regions in layer $m-1$ are passed to the hidden elements in the layer m, and the hidden elements in layer m are biased toward the feature information of spatial continuum sub-regions.

Take the retina as an example. As shown in Fig.3, units in the m-th layer have receptive fields of width 3 for the input retina. Thus, they are connected with 3 adjacent neurons in the m-th layer. Neurons in the $m+1$-th layer have similar connectivities as in m-th layer. The receptive field of the $m+1$-th layer is also 3. The excitation, which is generated by neurons outside the range of the previous receptive field, does not affect neurons in the current layer. Therefore, the architecture ensures that the strongest response can be produced by the learnt filters.

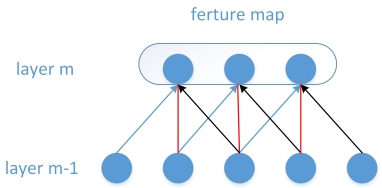

Figure 4. Share Weight Schematic

2) Shared Weights: Each filter h_i in a CNN is replicated across the entire visual field, shares the same parameters, and then **extracts** some features.

Fig. 4 shows the weight sharing. The three hidden neurons belong to the same feature map. The arrows with the same color have a same weight being shared and constrained. The original gradient descent method can be used to control the change of shared parameters.

Features are allowed to be detected since the neurons' replicating neglects their positions in the visual field. Additionally, reducing the number of freely learnt parameters improves the learning efficiency. The CNN can achieve a better generalization of vision problems due to the constraints on the model.

III. HYBRID METHOD FOR SHORT UTTERANCE SPEAKER RECOGNITION

The hybrid method for short utterance speaker recognition consists of three parts: 1) the self-adaptive GMM-MAP-UBM method, which is used for the feature alignment in the initial training of short speech; 2) the deep learning mechanism for short utterance speaker recognition, which is used for extracting fuller speaker characteristics; 3) the dual-judgment-mechanism for the fusion of the self-adaptive GMM-MAP-UBM and the deep learning decision.

A. The self-adaptive GMM-MAP-UBM method

The process of Maximum a Posteriori (MAP) criterion is defined as follows:

$$\eta^* = \arg\max_{\eta} p(\eta|X,\lambda_u) = \arg\max_{\eta} p(X|\eta,\lambda_u)p(\eta) \quad (16)$$

As same as EM algorithm, the iterative process consists of two parts. The first part is identical to the expectation step, which estimates the statistical parameters, such as n_i, $E_i(x)$, $E_i(x \cdot x')$ and $p\left(i|x^k\right)$. The training data are computed for each component in the UBM. Different from the second part of the EM algorithm, the new statistical estimates are mixed with the parameters which is from the UBM of the hybrid algorithm for the sake of better adaptation. The adapting equations are as follows:

$$\omega_i^* := [\alpha_i^\omega n_i/m + (1-\alpha_i^\omega)\omega_i] \cdot \gamma \quad (17)$$

$$\mu_i^* := \alpha_i^m E_i(x) + (1-\alpha_i^m) \cdot \mu_i \quad (18)$$

$$(\sigma_i^*)^2 := \alpha_i^v E_i(x \cdot x') + (1-\alpha_i^v)\left(\sigma_i^2 + u_i \cdot u_i'\right) - (u_i^*)^2 \quad (19)$$

We use Bayesian method to adapt the speech of the target speaker through the UBM parameters obtained in the early stage to derive speaker model. The $\{\alpha_i^\omega, \alpha_i^m, \alpha_i^v\}$ denote *weight*, *mean* and *variance* respectively. They are adaptation coefficients used to control the balance between new and old estimates. The best performance can be achieved by adapting the mean vectors.

As for the number of training samples, speaker special model may be able to deal with the phoneme information generated in training utterances, but fail to deal with the phonemes firstly appeared in testing utterances. To deal with the above problem, adapted GMM-MAP-UBM was proposed and shown in Fig.5 (d). In adapted GMM-MAP-UBM system, a single adaptation coefficient is used for all parameters ($\alpha_i^\omega = \alpha_i^m = \alpha_i^v = n_i/(n_i + r)$), which r = 14∼16. The accuracy of individual GMM models can be further improved by selecting the high probability mixture set as the initial training set of individual GMM training. The recognition process of the self-adaptive GMM-MAP-UBM method is shown in Fig.6.

B. The deep learning mechanism in hybrid method

The deep learning mechanism for short utterance speaker recognition is a preprocessed short speech spectrum and a convolution neural network model suitable for less sample training. The CNN has hidden layers and a complete connection layer of the perceptron unit. Former layers are consist of four layers (two convolution layers and two subsampling layers), the latter is used for the final classification.

The training procedure of CNN is as follows: firstly, we initialize N, L and M as the number of input neurons, middle neurons and output neurons respectively. Similarly, we define $X = (x_0, x_1, ..., x_N)$ as the input invector, $H = (h_0, h_1, ..., h_L)$ as the hidden layer's output vector, $Y = (y_0, y_1, ..., y_M)$ as the output vector, and $D = (d_0, d_1, ..., d_M)$ as the corresponding output label. V_{ij} presents the weight of input layer neuron i to the middle layer neuron j, and W_{jk} presents the weight of next two layers. Moreover, layers except input layer will be added threshold like θ_k and θ_j.

Using the input units, the result of the convolution operation of a set of filters is obtained as the feature maps. The result of the hidden layer is as follows.

$$h_j = f(\sum_{i=0}^{N-1} V_{ij}x_i + \phi_j) \quad (20)$$

Next, the feature maps of the subsampling layer are obtained by downsampling step. In the subsampling layer, the feature map remains holding the same features after pooling, but the size reduces to $1/n$ (assuming the pool size is n). The main effect of downsampling operation is to reduce feature dimensions and to a certain extent improve network robustness of displacement, zoom, and twist, which is shown in Eq.(21):

$$y_k = f(\sum_{j=0}^{L-1} W_u h_j + \theta_k) \quad (21)$$

In the above formula, $f(*)$ means activation function whose common form is like Eq.(22):

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2799928, IEEE Transactions on Industrial Informatics

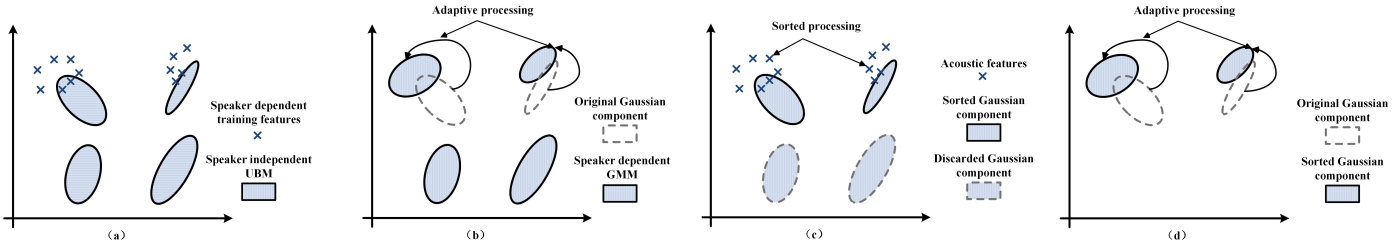

Figure 5. The example of adapted GMM-UBM training process

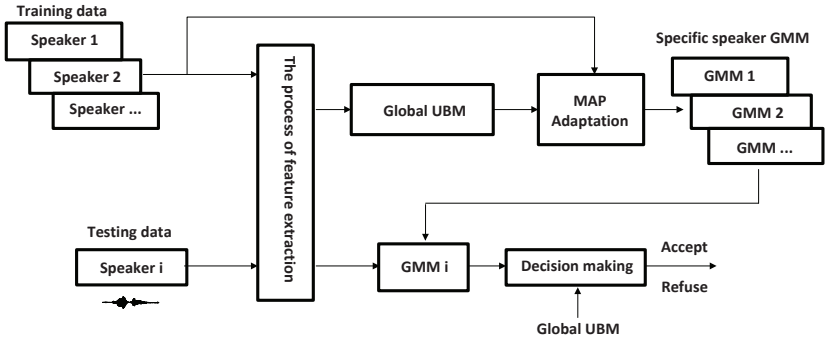

Figure 6. The recognition part of UBM-MAP-GMM

$$f(x) = \frac{1}{1 + e^{-kx}} \tag{22}$$

Now, we can show the whole process of the training procedure:

1) Randomly select N samples for training.
2) Randomly set all the weights and thresholds to a small value around zero to one and learning rate α, accuracy control parameter.
3) Confirm the ideal label D.
4) Calculate the output matrix H of the hidden layer and the final output vector Y of the output layer by using Eq.(20) and Eq.(21).
5) Calculate the cost function of the output layer:

$$\delta_k = (d_k - y_k)\, y_k\, (1 - y_k) \tag{23}$$

The cost function of the hidden layer is as follows:

$$\delta_j = h_j\, (1 - h_j) \sum_{k=0}^{M-1} \delta_k W_{jk} \tag{24}$$

6) The whole cost function is Eq.(25). We should make sure that the whole cost is smaller than accuracy control parameter, $E \leq \varepsilon$. If not, go to step 3 until the result is convergent.

$$E = \frac{1}{2} \sum_{k=0}^{M-1} (d_k - y_k)^2 \tag{25}$$

If the result is not satisfying, go to step 3, continue iteration until the result is convergent.

7) After each training, all parameters will be retained for the next usage.

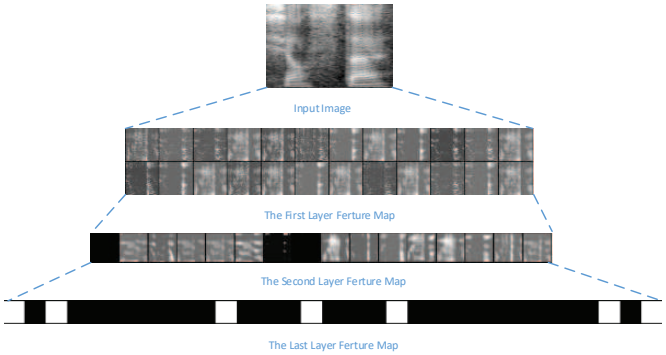

Figure 7. An example of feature maps

At last, learning all the weights of the filters is the aim of the training procedure, which is able to extract visual concepts from the raw image content, and obtains a suitable classifier. Low-level concepts such as edges and details can be typically identified by the first layer of the network, while the last layers are able to combine low-level features to identify complex visual concepts. Fig.7 shows that the examples of feature maps were obtained at each layer of a CNN. Former layers (top) identify simple structures such as edges and details, and next layers identify more complexly visual concepts. Finally, the last layer (bottom) acts as a classifier.

C. The dual-judgment-mechanism

Many experiments show that prosodic features are not suitable as a unique or independent feature used to identify speakers, although they contain useful identifiable information.

However, when the score falls into the double judgment area, the MFCC cannot effectively distinguish the speaker information according to the above observation. In this case, it is feasible to distinguish the speaker by extracting the prosodic information as the auxiliary distinguishing feature. Therefore, a number of voice features that are not obvious in the high partition effect can be introduced into the double judgment area to improve the recognition effect. In both noise and temporal robustness, the prosody feature can better describe the glottal information so that the prosodic feature may improve the recognition result.

Experiment results have been repeatedly observed and compared. In most cases, the short utterances from the hypothetical speaker and the short utterances of the impersonator have a significant score difference and the speaker score is much higher than the impersonator. However, when the scores were tightly distributed, the errors occur frequently. We believe that GMM as a standard model in the evaluation of high scores, the accuracy rate of the individual judgment is sufficient to meet the requirements, so second judgment is not necessary. When GMM score is not high, the CNN model makes the second decision. The miscarriage of justice can be reduced with the ability of CNN model refinement feature recognition.

The dual-judgement-mechanism works as follows: the first step is MFCC score evaluation. If the score is higher than the threshold, the MFCC score is the final score. If not, the prosodic feature score and CNN score are evaluated. The 3 value weighted mixed to obtain the final score, and the final score is used to recognized speaker. The formula for calculating the score is shown in Eq.26.

$$\Lambda(X) = \Lambda_0(X) + \gamma_1 \cdot \phi(p) + \gamma_2 \cdot \varphi(s) \tag{26}$$

Where $\Lambda_0(X)$ is the score of baseline MFCC, which is shown in Eq.27.

$$\Lambda_0(X) = \frac{1}{m} \cdot \sum_{k=1}^{m} [\log p(x^k|\lambda) - \log p(x^k|\lambda_u)] \tag{27}$$

where γ_1 and γ_2 denote the prosodic feature score and the CNN score. The p denotes the mean pitch extracted from the test utterance, and the s denotes the state of the CNN classification. The $\phi(p)$ denotes the similarity scores of the mean pitch, and the $\varphi(s)$ denotes the similarity scores of the result of CNN classification. We compare the MFCC score and the decision threshold in the first stage. The new score is calculated while the output in first-stage fails. In second stage, the MFCC score $\Lambda_0(X)$ is substituted to calculate the next score $\Lambda(X)$.

IV. EXPERIMENTS AND ANALYSIS

A. The DET Curves using GMM-MAP-UBM Model

Firstly, we describe the experimental corpus and its design ideas. We use adaptive GMM-MAP-UBM as our model. All the experiments are conducted based on a self-recording voice library which is recorded by 50 people. Each participant reads 20 phrases in short Chinese phrase, each of which is recorded

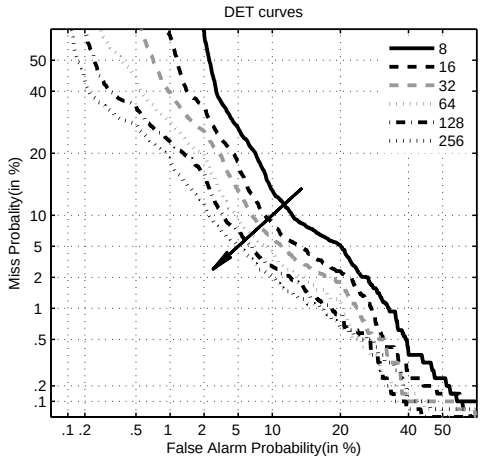

Figure 8. The DET curves about baseline GMM-MAP-UBM

10 times. Each short-utterance lasts about 1∼3 seconds. Our voice library is designed for text-related short-utterance speaker verification mission. Training data for specific GMM consists of 1∼2 minute utterances per speaker. The sampling rate is $16kHz$, $16bit$ quantization. At the same time, we use the same method to record short utterances from 50 different hypothesized speakers. Those attack results will be used to calculate FA and FR. We train specific GMM and UBM separately and independently. Training data for UBM consists of an hour of utterances. In this case, high order UBM was trained by pooling kinds of training data together. We trained specific GMM by adapting its parameters from same UBM using train utterances. It is not difficult to find that GMM and UBM have the same mixtures which are called system order under GMM-MAP-UBM architecture.

Then, we will describe how to confirm the threshold of speaker recognition system. If the threshold is used as the independent variable, the probability of FA and FR are taken as the dependent variable. We will get Receiver Operating Characteristic (ROC) curve. FRP and FAP are constrained to each other with the opposite trend, where the value of the intersection is the optimal threshold which is the minimum of error rate. Usually, we are more concerned about the error rate while ignoring the threshold. The next part of experiments validated the effect of model size on performance. UBMs that varies from $8 - 256$ are trained and evaluated by using the same data and methods as the last experiment. Eventually, the Fig.8 shows DET curves of different model orders. Within these DET curves, it is not difficult to find that the best point in the curve is around 128 and 256 mixtures. Specifically, the best EER of system is 4.9%.

B. The Extraction of CNN

We put forward speech recognition model based on phonological map deep learning, shown in Fig.9. This model can exploit the advantage of deep CNN and extract feature maps from the speech images, which has a higher recognition rate. The algorithm incorporates two phases: phases training and testing. The image is normalized to 256×256 in the training

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2799928, IEEE Transactions on Industrial Informatics

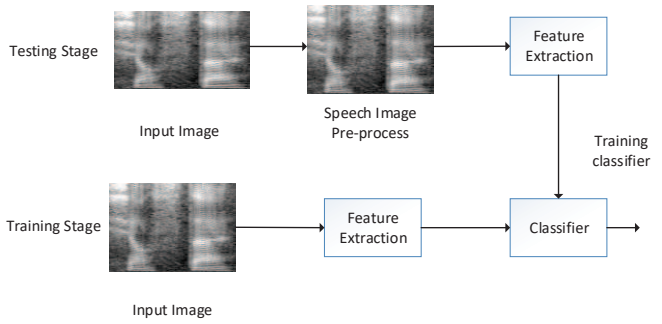

Figure 9. The process of speech identification

phase and then input into the deep CNN to extract its feature and train classifier. In the test phase, we take the processed image and the extracted CNN feature as input, and then classify and identify them.

Experimental images are from the speed spectrum images of the 50 individual speed samples database, choosing 50 people as the training samples. Each sample remains 10 speech spectrum pictures. We use 8 samples for training and 2 for testing of each person. The initial sound spectrograph size is close to 300×200.

The CNN model we used consists of 7 hidden layer, and each layer adopt "s" activation function. Firstly, the input layer will adjust the input image to 256×256. After the input layer, the image will convolute with a 11 size kernel. And in the convolution layer we can obtain 96 feature maps. We choose 5000 as an appropriate number of iterations in this experiment, which can get the appropriate cost.

Table.I shows the relationship between the iteration number and the training loss & the test accuracy. Because the relationship between the number of iterations and accuracy and loss rate is nonlinear, we take the average of 100 tests before and after the test point as the test value at that point. For example, 1000 iterations have a mean of 100 detections over 950∼1050 iterations. It can be inferred from the experimental, due to non-linearity, choosing the optimal iteration number is an important thing. The last recognition rate only using CNN in this experiments is about 90%.

Table I
THE RELATIONSHIP BETWEEN THE LOSS & ACCURACY AND THE NUMBER OF ITERATIONS

Iterations	500	1000	2000	3000	4000	5000
Train loss	1.71	0.27	0.12	0.02	0.06	0.01
Test Accuracy	0.47	0.65	0.72	0.75	0.81	0.87

C. The DET Curves under Dual-judgment-mechanism

Our experiments show the performance on the effect of the proposed system under dual-judgment-mechanism varied from mixtures. The prosodic acoustic features and dual-judgment-mechanism are adopted. Fig.10 shows DET curves in detail. Specifically, the best EER of the system is 2.5%. That is to say,

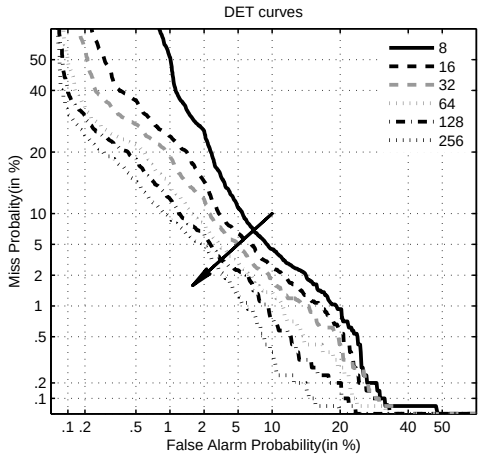

Figure 10. The DET curves about the new model

our new algorithm can help reduce the EER and thus improve the recognition accuracy to a certain extent.

V. CONCLUSION

In this paper, a hybrid model of adaptive GMM and deep learning for short utterance is proposed. By improving the alignment of the initial training data of the EM algorithm, it improves the recognition accuracy of the GMM model for short utterance. At the same time, by training the preprocessed speech spectrum in deep network to extract the deep features of the full frequency spectrum of short utterance, it achieves effective training and recognition for a small number of biometric samples. Finally, a dual decision mechanism is proposed based on this hybrid model, which improves the accuracy of short utterance recognition significantly. In the future, we will further improve the accuracy and richness of feature extraction in short utterance and speech.

ACKNOWLEDGMENT

This research is supported by National Natural Science Foundation of China (Nos. 61672300, 61772162, No. 61472091), National Natural Science Foundation of Tianjin (No. 16JCYBJC15500), Joint fund of National Natural Science Fund of China (No. U1709220), National Key Research and Development Program of China (No. 2016YFB0800201), Natural Science Foundation of Guangdong Province for Distinguished Young Scholars (2014A030306020), Guangzhou scholars project for universities of Guangzhou (No. 1201561613), Science and Technology Planning Project of Guangdong Province, China (2015B010129015), National Natural Science Foundation for Outstanding Youth Foundation (No. 61722203), and Zhejiang Science Fund (NO.LY16F020016).

REFERENCES

[1] Z. Wu, Z. Yu, J. Yuan, and J. Zhang, "A twice face recognition algorithm," *Soft Computing*, vol. 20, no. 3, pp. 1007–1019, 2016.
[2] J. P. Campbell, "Speaker recognition: A tutorial," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1437–1462, 1997.

[3] R. Vogt, S. Sridharan, and M. Mason, "Making confident speaker verification decisions with minimal speech," *IEEE transactions on audio, speech, and language processing*, vol. 18, no. 6, pp. 1182–1192, 2010.

[4] P. Kenny, G. Boulianne, and P. Dumouchel, "Eigenvoice modeling with sparse training data," *IEEE transactions on speech and audio processing*, vol. 13, no. 3, pp. 345–354, 2005.

[5] K. Zinchenko, C.-Y. Wu, and K.-T. Song, "A study on speech recognition control for a surgical robot," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 607–615, 2017.

[6] M. Nosratighods, E. Ambikairajah, J. Epps, and M. J. Carey, "A segment selection technique for speaker verification," *Speech Communication*, vol. 52, no. 9, pp. 753–761, 2010.

[7] R. J. Vogt, B. J. Baker, and S. Sridharan, "Factor analysis subspace estimation for speaker verification with short utterances," *Interspeech*, 2008.

[8] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Interspeech*, 2011, pp. 249–252.

[9] Y. Jiang, H. Song, R. Wang, M. Gu, J. Sun, and L. Sha, "Data-centered runtime verification of wireless medical cyber-physical system," *IEEE transactions on Industrial Informatics*, vol. 13, no. 4, pp. 1900–1909, 2017.

[10] P. Li, J. Li, Z. Huang, C.-Z. Gao, W.-B. Chen, and K. Chen, "Privacy-preserving outsourced classification in cloud computing," *Cluster Computing*, vol. 4, no. 510, pp. 1–10, 2017.

[11] J. Li, X. Huang, J. Li, X. Chen, and Y. Xiang, "Secure attribute-based data sharing for resource-limited users in cloud computing," *Computers and Security*, vol. 72, no. 1, pp. 1–12, 2018.

[12] Z. Pan, J. Lei, Y. Zhang, X. Sun, and S. Kwong, "Fast motion estimation based on content property for low-complexity h. 265/hevc encoder," *IEEE Transactions on Broadcasting*, vol. 62, no. 3, pp. 675–684, 2016.

[13] B. Tang, X.-j. Gong, W. Wei, H. Wang *et al.*, "Intelligent fault diagnosis of the high-speed train with big data based on deep neural networks," *IEEE Transactions on Industrial Informatics*, 2017.

[14] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[15] F. Richardson, D. Reynolds, and N. Dehak, "Deep neural network approaches to speaker and language recognition," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1671–1675, 2015.

[16] C. Yuan, X. Li, Q. J. Wu, J. Li, and X. Sun, "Fingerprint liveness detection from different fingerprint materials using convolutional neural network and principal component analysis," 2017.

[17] Y. Li, G. Wang, L. Nie, Q. Wang, and W. Tan, "Distance metric optimization driven convolutional neural network for age invariant face recognition," *Pattern Recognition*, vol. 75, pp. 51–62, 2018.

[18] Y. Gao, X. Shan, Z. Hu, D. Wang, Y. Li, and X. Tian, "Extended compressed tracking via random projection based on msers and online ls-svm learning," *Pattern Recognition*, vol. 59, pp. 245–254, 2016.

[19] Y. Li, Z. Peng, D. Liang, H. Chang, and Z. Cai, "Facial age estimation by using stacked feature composition and selection," *The Visual Computer*, vol. 32, no. 12, pp. 1525–1536, 2016.

[20] K. Wang, D. Zhang, Y. Li, R. Zhang, and L. Lin, "Cost-effective active learning for deep image classification," *IEEE Transactions on Circuits and Systems for Video Technology*, 2016.

[21] S. Hershey, Chaudhuri *et al.*, "Cnn architectures for large-scale audio classification," in *ICASSP*. IEEE, 2017, pp. 131–135.

[22] Z. Wu, B. Liang, L. You, Z. Jian, and J. Li, "High-dimension space projection-based biometric encryption for fingerprint with fuzzy minutia," *Soft Computing*, vol. 20, no. 12, pp. 4907–4918, 2016.

[23] Z. Wang, "Unsupervised recognition and characterization of the reflected laser lines for robotic gas metal arc welding," *IEEE Transactions on Industrial Informatics*, 2017.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[25] D. Cireşan, U. Meier, J. Masci, and J. Schmidhuber, "Multi-column deep neural network for traffic sign classification," *Neural Networks*, pp. 333–338, 2012.

[26] Z. Huang, M. Dong, Q. Mao, and Y. Zhan, "Speech emotion recognition using cnn," in *ACM International Conference on Multimedia*, 2014, pp. 801–804.

[27] G. Tian, "Hybrid genetic and variational expectation-maximization algorithm for gaussian-mixture-model-based brain mr image segmentation," *IEEE transactions on information technology in biomedicine*, vol. 15, no. 3, pp. 373–380, 2011.

Zheli Liu received the BSc and MSc degrees in computer science from Jilin University, China, in 2002 and 2005, respectively. He received the PhD degree in computer application from Jilin University in 2009. After a postdoctoral fellowship in Nankai University, he joined the College of Computer and Control Engineering of Nankai University in 2011. Currently, he works at Nankai University as an Associate Professor. His current research interests include applied cryptography and data privacy protection.

Zhendong Wu received the M.S. degree and the PhD degree in Computer Science and Technology from the Zhejiang University, Hangzhou, China, in 2004 and 2007, respectively. Currently, he is an Associate Professor with the School of Cyberspace, Hangzhou Dianzi University, Hangzhou, China. His current research interests include biometrics, biological cryptography, machine intelligence and natural language research.

Tong Li received his B.S. and M.S. from Taiyuan University of Technology and Beijing University of Technology, in 2011 and 2014, respectively, both in Computer Science & Technology. He got his Ph.D degree in information security from Nankai University at 2017. Currently, he is a post-doctoral research at Guangzhou University. His research interests include applied cryptography and data privacy protection in cloud computing.

Jin Li is currently a professor and vice dean of School of Computer Science, Guangzhou University. He received his B.S. (2002) and M.S. (2004) from Southwest University and Sun Yat-sen University, both in Mathematics. He got his Ph.D degree in information security from Sun Yat-sen University at 2007. His research interests include design of secure protocols in Cloud Computing (secure cloud storage and outsourcing computation) and cryptographic protocols. He served as a senior research associate at Korea Advanced Institute of Technology (Korea) and Illinois Institute of Technology (U.S.A.) from 2008 to 2010, respectively. He has published more than 100 papers in international conferences and journals. His work has been cited more than 7620 times at Google Scholar and the H-Index is 34. He received three National Science Foundation of China (NSFC) Grants, including NSFC Outstanding Youth Foundation.

Chao Shen is currently an Associate Professor in the School of Electronic and Information Engineering, Xian Jiaotong University of China. He is also with the Ministriy of Education Key Lab for Intelligent Networks and Network Security. He was a research scholar in Carnegie Mellon University from 2011 to 2013. His research interests include network security, human computer interaction, insider detection, and behavioral biometrics.