IMPROVED DESIGN OF DTW AND GMM CASCADED ARABIC SPEAKER VERIFICATION

Shuoshuo Chen, Junbo Zhao and Ruiqi Yang

School of Electronic Information
Wuhan University, Wuhan, China

Email: sschen@whu.edu.cn

Abstract

In this paper, we discuss about the design, implementation and assessment of a two-stage Arabic speaker recognition system, which aims to recognize a target Arabic speaker among several people. The first stage uses improved DTW (Dynamic Time Warping) algorithm and the second stage uses SA-KM-based GMM (Gaussian Mixture Model). MFCC (Mel Frequency Cepstral Coefficients) and its differences form, as acoustic feature, are extracted from the sample speeches. DTW provides three most possible speakers and then the recognition results are conveyed to GMM training processes. A specified similarity assessment algorithm, KL distance, is applied to find the best match with the target speaker. Experimental results show that text-independent recognition rate of the cascaded system reaches 90 percent.

Keywords: *Arabic speaker, multi-fold MFCC, improved GMM, verification*

Abstrak

Dalam paper ini, kami membahas desain, implementasi dan penilaian sistem pengenalan dua tahap untuk penutur Bahasa Arab, yang bertujuan untuk mengenali target penutur Bahasa Arab di antara beberapa orang. Tahap pertama menggunakan *algoritma improved DTW (Dynamic Time Warping)* dan tahap kedua menggunakan SA-KM berbasis GMM *(Gaussian Mixture Model)*. MFCC *(Mel Frequency Cepstral Coefficients)* dan variasi perbedaannya, seperti fitur akustik, diekstrak dari sample suara. DTW menyediakan tiga penutur yang paling mungkin dan kemudian hasil pengenalan diteruskan ke proses pelatihan GMM. Sebuah algoritma penilaian kesamaan yaitu *KL distance*, diaplikasikan untuk menemukan pasangan yang paling cocok dengan penutur sasaran. Hasil penelitian menunjukkan bahwa tingkat pengenalan teks-independen dari sistem mencapai 90 persen.

Kata kunci: *penutur Bahasa Arab, multi-fold MFCC; improved GMM; verifikasi*

1. Introduction

Speaker recognition, also known as voiceprint recognition, analyzes speaker's voice for the purpose of speaker identification or verification. From the late 1970s to the late 1980s, the speaker recognition research was focused on the acoustic parameters and pattern matching methods. Steven B.Davis was the first one to propose the concept of MFCC[1]. Soon it became a mainstream speaker recognition parameter. At this time, dynamic time warping (DTW)[2] proposed by Itakura, vector quantization (VQ)[3] by Gray, hidden Markov model (HMM)[4] by Leonard E. Baum, artificial neural network (ANN)[5] by Zeidenberg and other technologies had been widely applied and become the core technologies of speaker recognition. After 1990s, when Reynolds elaborated GMM[6], it has quickly become the current mainstream technology in machine learning.

For related work, Markov proposed a new speaker identification system based on GMM[7], where the likelihood normalization technique is widely used for speaker verification. Sturim presented an approach to close the gap between text-dependent and text-independent speaker verification performance[8]. Pellom presented a novel algorithm for reducing the computational complexity of identifying a speaker within a Gaussian mixture speaker model framework[9].

In this paper, the recognition system comprises revised DTW stage, SA-based K-means clustering sub-stage, GMM stage and similarity

assessment sub-stage. Speeches of 10 Arabic speakers are collected to set up a corpus. Pre-treated voice signals are considered as standard test or reference templates and then sent into the cascaded system for recognition. An overview block diagram is shown in Figure 1.

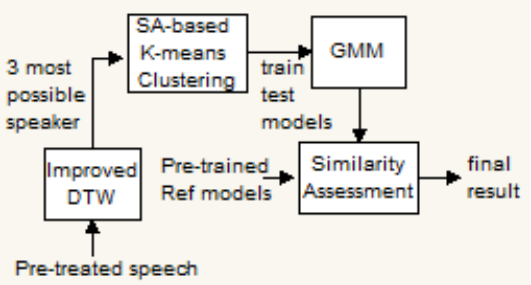

Fig 1. Cascaded System Block Diagram

2. Feature Extraction

Arabic is a Semitic language, standard Arabic has 34 basic phonemes, of which six are vowels and 28 are consonants. There are many features that distinguish Arabic from other languages. Its diacritic symbols consisting of short vowels which are normally invisible. This is because Arabic alphabets contain letters for long vowels and consonants. At the same time, short vowels and consonants can be merged according to Arabic grammar as the Arabic-texts are almost never fully diacritic.

Since the voice signal is a typical non-stationary signal, combined with the influence of respiratory airflow, external noise and current interference, the speech signal cannot be directly used to extract the feature. All of them will be processed by endpoint detection program and remain no more silence. Therefore, the pre-treated speech signals can be used for feature extraction. Feature parameters properties directly affect the system performance and efficiency. The most widely used parameters are LPC, LPCC and MFCC.

2.1 Linear Prediction Coefficients (LPC)

Vocal tract properties can be molded by using all-pole model with the help of LPC features. These features represent the main vocal tract resonance property in the acoustic spectrum. Each speech has its own format structure. It is the major difference between the different speeches. LPC highlights these formant structures for speech to make differentiation between them. Each LPC is independent with others in pitch and intensity. The extraction method can be found in John E. Markel's work[10].

2.2 Mel-Frequency Cepstral Coefficients

The Mel-Cepstrum makes use of the auditory system principle, it has high discriminating power at lower frequencies compared to higher frequencies. Cepstral coefficients are the mostly used features in speaker recognition due to many reasons, the most important one is good performance in representing vocal tract changes, capable to contend with convolution channel distortion and robust against noise. The extraction method can be found in Steven B.Davis's work [1].

2.3 Differential MFCC

The standard cepstrum coefficient MFCC only reflects the static feature of the voice. The dynamic feature can be described by the differential spectrum of these static features. In the following, some experiments will show that combining the dynamic with static feature contributes to improvement of the recognition performance effectively.

Different parameters and their combing forms are applied for text-independent recognition, the abbreviation used are:

1. Linear Prediction Cepstrum Coefficient (LPCC)
2. Mel-frequency Cepstral Coefficients (MFCC)
3. First order differential MFCC (ΔMFCC)
4. Second order differential MFCC ($\Delta\Delta$MFCC)

The result is shown in Figure 2.

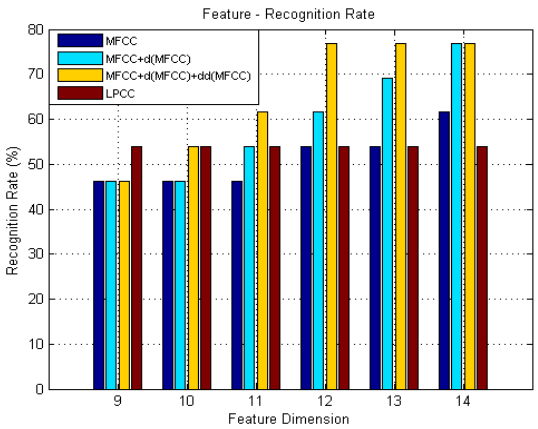

Fig 2. Recognition result of different parameters

The recognition rate of LPCC does not increase from 9 orders to 14 orders. Since LPCC reflects the sound model, it is not suitable for speaker recognition. The recognition rate of MFCC is slowly rising with increasing orders

while an obvious increase can be seen after adding first order differential MFCC and the most obvious upward appears after adding second order differential MFCC. Besides, the recognition rate of ΔΔMFCC no longer rises after 12 orders, this may caused by insufficient corpus for training. Since 14-dimensional ΔMFCC and ΔΔMFCC are beyond the computing power of the experiment platform, 12 orders combination (MFCC+ΔMFCC+ΔΔMFCC) is chosen as the feature.

3. Dynamic Time Warping

Dynamic time warping is an algorithm for measuring similarity between two sequences which may vary in time or speed. Traditional DTW saves a big array which is not the arbitrary choice of the path. Considering the actual situation of pronunciation, although the voice is different when it goes faster and slower, the order of each part is not possible to be reversed. So the path must start from the bottom left corner to the upper right corner of the end. To prevent the blind search, the improved DTW usually does not permit sub-sloping path. Maximum slope is designated as 2 and the minimum slope at 1/2. The restricted search path is shown in Figure 3.

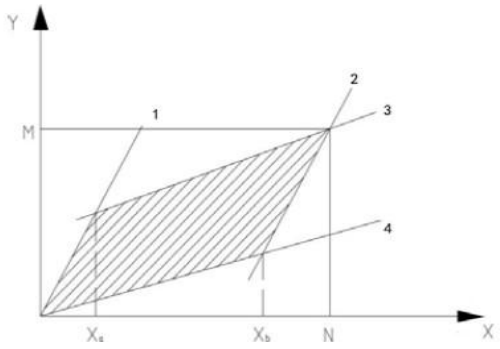

Fig 3. Restricted search path of improved DTW

The purpose of using DTW as first stage is to reduce the computation complexity of the second stage. Because DTW is an almost real time algorithm while GMM takes a lot of time training models. DTW can provide a cursory filtering and keep a few possible speakers remaining. The accurate matching will be accomplished by GMM.

4. Gaussian Mixture Model

4.1. GMM Principle

Gaussian mixture model (GMM) is an effective tool for data modeling and pattern classification. GMM assumes the data under modeling is generated via a probability density distribution which is the weighted sum of a set of Gaussian PDF. To train a GMM is to calculate a set of Gaussian PDF and make the weighted set similar to the feature.

4.2. K-means Clustering

K-means clustering is a method of cluster analysis which aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean. According to Reference X, the algorithm proceeds by alternating between two steps:

1. *Assignment step*: Assign each observation to the cluster whose mean is closest to it as given in equation (1).

$$S_i^{(t)} = \{x_p : \| x_p - m_i^t \| \le \| x_p - m_j^t \| \ \forall 1 \le j \le k\} \quad (1)$$

2. *Update step*: Calculate the new means to be the centroids of the observations in the new clusters as given in equation (2).

$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j \quad (2)$$

The algorithm converges when the assignments of x do not vary any more.

The initial value of each parameter is selected arbitrarily during GMM training, the final value is found through iterative convergence. The shortage is that this approach requires a lot of iterations. If a more ideal cluster center can be selected in advance, the computational complexity of GMM training will efficaciously reduce. So K-means clustering algorithm is introduced for getting initial value. From a comparative trial, the advantage of K-means in GMM training is shown in Table I.

TABLE I
RANDOM INITIAL VALUE VS. K-MEANS BASED INITIAL VALUE

	Random	K-means
Iterations	104	75

As we can see, choosing the results after K-means clustering as initial values to train GMM can reduce the amount of computation by 27.88%.

4.3. Simulated Annealing

Simulated Annealing (SA) refers to a process that particle motion state in solids will be changed by temperature. The solid is heated to high temperature, which makes intrinsic energy increase as well as particles accelerated by the temperature, and then the particle motion become disorderly. However, it gradually turns out to be orderly when cooling down.

The probability is when particle motion approach steady at T℃ in accordance with the principle of Metropolis. Where E is internal

energy at temperature *T*, standing for changes of internal energy. *K* is Boltzmann constant. Solution combinatorial optimization of SA algorithm can be obtained from objective function f by stimulating internal energy E and control parameter described with temperature *T*. If material state is defined by internal energy of particle, Metropolis algorithm can be described as the annealing process with a simply mathematical model. On the assumption that internal energy is shaped by the material at the status of i , the material should be abode by following rule of changing at the state from i to j at temperature *T*. Briefly, this process can be seen below:

If $E(j) \leq E(i)$: the state change is accepted.
If $E(j) > E(i)$: the state change is accepted by the probability p

$$p = e^{(E(i)-E(i))/(kT)} \tag{3}$$

SA algorithm is composed with solution space, objective function and initial solution. SA algorithm begins to calculate with initial solutions and initial value of control parameters. The iterative process, which is executed by producing new solution→calculating objective function→judging→accepting or discarding, is repeated to current solution. By random search with the probabilistic jumping property and repeat sampling with temperature drop, global optimization solution can be found finally.

In the experiment, there is a 36-order feature vector to be trained by GMM. In order to reflect the capability of SA algorithm to avoid falling into local optimum, five significant "noise" (amplitude of 1000) is added to the feature parameters. Results are shown in Table II and Table III.

TABLE II
THE MAXIMUM CLUSTERING MEAN WITH SA-BASED K-MEANS

Vector Dimension Number	Maximum mean
4	1.5225
13	4.3070
15	2.5707
36	0.4184

Four obvious noise polluted means of the total five are found in the result of clustering mean with normal K-means. On the other side, no obvious noise is found in the result of clustering mean with SA based K-means. SA algorithm has shown its superiority in finding global optimum and avoiding noise pollution.

TABLE III
(RIGHT) THE MAXIMUM CLUSTERING MEAN WITH K-MEANS CLUSTERING

Vector Dimension Number	Maximum mean
4	1000
13	1000
15	1000
36	1000

5. Similarity Assessment

5.1. Bhattacharyya Distance

The Bhattacharyya distance of the two probability distributions is defined in equation (4):

$$d_B(p_1, p_2) = -\ln\left(\int_{R^n} \sqrt{p_1(x)p_2(x)}dx\right) \tag{4}$$

in which p_{a_i} and p_{b_i} denote two proprobabilistic Gaussian mixture models. Furthermore, our method concerns more about the difference between each pair of the Gaussian components rather than the distinction of the entire mixture models. Therefore, we obtain the distance concretely by equation (5):

$$d_{\text{Bhattacharyya}} = \sum_{i=1}^{M} d_B(p_{a_i}, p_{b_i}) \tag{5}$$

in which p_{a_i} denotes the trained GMM model and p_{b_i} is the testing GMM model.

Figure axis labels are often a source of confusion. Use words rather than symbols. As an example, write the quantity "Magnetization," or "Magnetization *M*," not just "*M*" Put units in parentheses. Do not label axes only with units. As in Fig. 1, for example, write "Magnetization (A/m)" or "Magnetization ($\mathrm{A \cdot m^{-1}}$)," not just "A/m." Do not label axes with a ratio of quantities and units. For example, write "Temperature (*K*)," not "Temperature/K." Multipliers can be especially confusing. Write "Magnetization (kA/m)" or "Magnetization (10^3A/m)." Do not write "Magnetization (A/m) × 1000" because the reader would not know whether the top axis label in Fig. 1 meant 16000 A/m or 0.016 A/m. Figure labels should be legible, approximately 8 to 12 points type.

5.2. Kullback-Leibler Distance

The KL distance is an information theoretic distance measure between probability density functions. It could be acquired by equation (6):

$$d_{KL}(p_1, p_2) = \int p_1(x) \log \frac{p_1(x)}{p_2(x)} dx \qquad (6)$$

which measures the distance between two different distributions. Since this distance is not symmetric apparently, we modify it by equation (7):

$$d_{sKL} = d_{KL}(p_1, p_2) + d_{KL}(p_2, p_1) \qquad (7)$$

Comparison between Bhattacharyya distance and KL distance will be given in Figure 4.

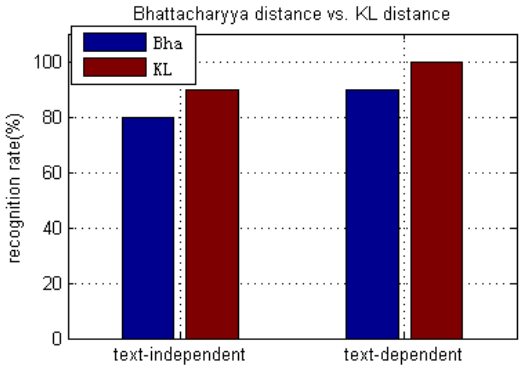

Fig 4. Bhattacharyya distance vs. KL distance

Accordingly, the Bhattacharyya distance provides 80% and 90% recognition rate in irrelevant test and relevant test relatively and KL distance reaches 90% and 100% correspondingly under this experimental condition. We can know that KL distance renders Gaussian mixture models a more efficient measurement.

6. Experimental Results

DTW and GMM are cascaded to elicit the integral recognition rate. The DTW stage chooses 3 most possible speakers. Then the GMM stage finds the most matching one from those 3 speakers. Both the DTW and GMM stage share the same corpus and feature. The outcome is listed in Table IV.

Comparatively speaking, text-dependent recognition is more accurate than text-independent recognition. Considering the time consuming, the first stage only takes about 9 seconds to make a rough filtering and the second stage takes about 6 minutes. If DSPs are used, the time spent will be further reduced. So, we believe the design has met the requirements of this recognition system.

TABLE IV
INTEGRAL RECOGNITION OF TEXT-INDEPENDENT AND TEXT-DEPENDENT

	Text Independent	Text Dependent
Recognition Rate	90%	100%
Time Consuming (DTW)	9.016 sec	9.227 sec
Time Consuming (GMM)	6 min	6 min

7. Conclusion

In this paper, a two-stage Arabic speaker recognition system is introduced for recognizing a target Arabic speaker from several people. The cascaded system uses improved DTW (Dynamic Time Warping) algorithm in the first stage and SA-KM-based GMM (Gaussian Mixture Model) in the second stage. MFCC (Mel Frequency Cepstral Coefficients) and its differences are extracted to serve as acoustic feature. A specified algorithm entitled KL distance is applied as similarity assessment. The integral recognition rate of text-independent recognition is up to 90 percent. For further work, we will pay more attention to BP neural network and random forests in unsupervised machine learning.

References

[1] Davis, Steven, and Paul Mermelstein. "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences." Acoustics, Speech and Signal Processing, IEEE Transactions on 28.4 (1980): 357-366.

[2] Itakura, Fumitada. "Minimum prediction residual principle applied to speech recognition." Acoustics, Speech and Signal Processing, IEEE Transactions on 23.1 (1975): 67-72.

[3] Gray, Robert. "Vector quantization." ASSP Magazine, IEEE 1.2 (1984): 4-29.

[4] Baum, Leonard E., et al. "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains." The annals of mathematical statistics 41.1 (1970): 164-171.

[5] Zeidenberg, Matthew. Neural networks in artificial intelligence. Ellis Horwood, 1990.

[6] Reynolds, Douglas A., Thomas F. Quatieri, and Robert B. Dunn. "Speaker verification using adapted Gaussian mixture models." Digital signal processing 10.1 (2000): 19-41.

[7] Markov, Konstantin, and Seiichi Nakagawa. "Frame level likelihood normalization for text-independent speaker identification using Gaussian mixture models." Spoken Language, 1996. ICSLP 96. Proceedings.,

Fourth International Conference on. Vol. 3. IEEE, 1996.

[8] Sturim, Douglas E., et al. "Speaker verification using text-constrained Gaussian mixture models." Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on. Vol. 1. IEEE, 2002.

[9] Pellom, Bryan L., and John HL Hansen. "An efficient scoring algorithm for Gaussian mixture model based speaker identification." Signal Processing Letters, IEEE 5.11 (1998): 281-284.

[10] Markel, John E., and Augustine H. Gray. Linear prediction of speech. Springer-Verlag New York, Inc., 1982.