UNIVERSAL BACKGROUND MODELS FOR REAL-TIME SPEAKER CHANGE DETECTION

Ting-Yao WU[1] **Lie LU** **Ke CHEN** **Hong-Jiang ZHANG**

Peking University *Microsoft Research Asia* *Birmingham University* *Microsoft Research Asia*
Beijing, China, 100871 *Beijing, China, 100080* *Birmingham, UK* *Beijing, China, 100080*
tywu@cis.pku.edu.cn *llu@microsoft.com* *K.Chen@cs.bham.ac.uk* *hjzhang@microsoft.com*

This paper addresses the problem of real-time speaker change detection in TV news broadcast, in which no prior knowledge on speakers is assumed. To remove the unreliable frames and background frames in the speech stream, we propose a new approach for feature categorization based on Gaussian Mixture Model - Universal Background Model (GMM-UBM). The feature vectors are categorized into three sets, which include reliable speech, doubtful speech and unreliable speech. Then a novel distance measure is presented correspondingly for real-time speaker change detection. Extensive experiments demonstrate its good performance, and intrinsic difficulties on real-time speaker change detection are discussed as well in this paper.

1 Introduction

With the rapid increase of the amount of information, the need for storing, classifying and indexing information database is highly demanded. Many methods have been proposed to manage different database. For a speech database, one useful tool is to classify and index the speech based on its speaker identities. In many applications, such as live net-meeting and real-time conversation, the further demand is to segment and track speech based on speakers in real time.

Classifying and indexing the speech stream consists of two steps. The first step is to find the speaker change points in a speech stream, which can be called as speaker segmentation or speaker change detection. The second step is to identify the new coming speaker once a speaker change is detected, which is named as speaker tracking.

Speaker segmentation and tracking are highly associated with the traditional speaker recognition. But speaker segmentation and tracking are more difficult than speaker recognition. In general, in a speaker recognition system, speaker models are usually well trained. But in real-time speaker segmentation and tracking system, there is no prior knowledge on speakers, including speaker identities and the number of speakers. Thus, no data can be achieved to train appropriate models for speakers *a priori*. On the other hand, in speaker segmentation and tracking, training and testing data have no obvious boundary, whereas acoustic data have often been labeled in speaker recognition.

In this paper, we focus on real-time speaker segmentation in TV broadcast news. Our goal is to detect potential speaker change points in a speech stream in real time and to segment the stream into homogeneous speaker clips. That means,

[1] This work was done when this author was a visiting student at Microsoft Research Asia, Media Computing Group.

what we care is the change of speaker identities, not the change of background or channel conditions. Several approaches have been proposed for speaker change detection. Chen [2] and Delacourt [9] presented an approach to detect changes with speaker identities, environment conditions and channel conditions using Bayesian Information Criterion (BIC). Mori [4] addressed the problem of speaker change detection and speaker tracking, where the speaker grouping information was used in speaker adaptation for speaker recognition. Wilcox [3] proposed to use the Hidden Markov Model (HMM) for segmentation of conversational speech based on speaker identities. Couvreur [7] employed the "*Chop – and - Recluster*" method to build an automatic system for speaker-based segmentation of broadcasting news. The aforementioned work tried to solve the problem of segmenting the audio stream into homogeneous clusters in terms of speaker identity. However, many speaker segmentation systems cannot be operational in real time since iterative computation is inevitably involved in most of those approaches.

In our former work [8], we established a speaker segmentation system based on the discriminative distance between every two adjacent windows. We estimated a speaker model for each window. However, it is unavoidable that there exist some non-speech frames in such a window, which makes the estimated speaker model not so accurate. Thus, it is necessary to find an approach to categorize the features into speech part and non-speech part, so that we can grasp the speaker's characteristics. Beigi [6] applied K-Means algorithm to categorize different features in a window into three classes, which includes silence or background-related features, speech-related features and speaker-related features. However, these three classes are very difficult to discriminate, and K-Means only clusters the feature vectors into three sets but could not tell which cluster belongs to which set. Moreover, K-Means algorithm unavoidably needs iterative operations, which might prohibit the applicability of this method in a real-time task. To solve these problems, we propose a new approach of feature categorization based on universal background model (UBM), which can discriminate reliable speech more clearly and is suitable for real-time processing.

The idea of UBM has been proposed for many years in order to improve speaker recognition system against mismatch. Recently, Reynold [5] used GMM-UBM in speaker model normalization and adaptation, which leads to the better performance especially as training data is limited. GMM-UBM is a large speaker-independent GMM trained by pooling plenty of speech data by the expectation - maximization (EM) algorithm, or by pooling the subpopulation models trained by individual UBMs [5].

Unlike the previous work, we employ the UBM idea to categorize the features into three parts, which include reliable speech, doubtful speech and unreliable speech, in one window. Base on the categorization results, a novel distance measure is proposed. It enhances the effects from speaker change, and decreases the influence of channel change. Experiments showed it work better than other distance measure in our speaker segmentation system.

The rest paper is organized as follows. The overview of our system is describes in Section 2. Section 3 discusses our approach on feature categorization and distance measure in detail. Section 4 gives the experimental results on the Hub-4 broadcast TV news, followed by a discussion of some issues.

2 System Overview

The flow chart of our proposed real-time speaker segmentation is illustrated in Figure 1. It consists of three modules: front-end processing, feature vectors categorization and speaker segmentation. The input stream is first pre-segmented into 3-second windows with 2.5-second overlapping. Each window is pre-processed by removing silence frames using the simple energy threshold. In the step of feature categorization, UBM is used to categorize the features into three parts and select the reliable speech part. Then, the distance between two reliable speech parts in every two adjacent windows is measured. Local peak is found in such a distance series, and it is considered as potential speaker change boundary if it is larger than thresholds.

2.1 Front-end processing

The input audio stream is first down-sampled into a uniform format: 8KHZ, 16bits, mono channel, whatever the input format is. The speech stream is then pre-emphasized and divided into sub-segments by 3-second window with 2.5-second overlapping. That is, the basic processing unit is 3-second and the temporal resolution of the segmentation is 0.5 second. The sub-segment is further divided into non-overlapping 25ms-long frames. From each frame, 10-order LSP and 16-order MFCC are both extracted, since LSP and MFCC represent the different characteristics of speaker.

2.2 Speaker Segmentation

The general idea of real-time speaker segmentation is to find the local maximal peak in the distance series calculated from two adjacent windows. Thus, a discriminative distance measure is crucial for speaker segmentation. We will discuss it in the following section.

After the distance is computed between each two neighboring windows at each time slot, as the Fig 2 illustrates. A potential speaker change point is found between *i*th and (*i*+1)th window, if the following conditions are satisfied:

$$\begin{aligned} &D(i,i+1) - D^{i}_{\min,left} > \theta_i \,, \\ &D(i,i+1) - D(i+1,i+2) > 0 \,, \\ &D(i,i+1) > Th_i \end{aligned} \tag{1}$$

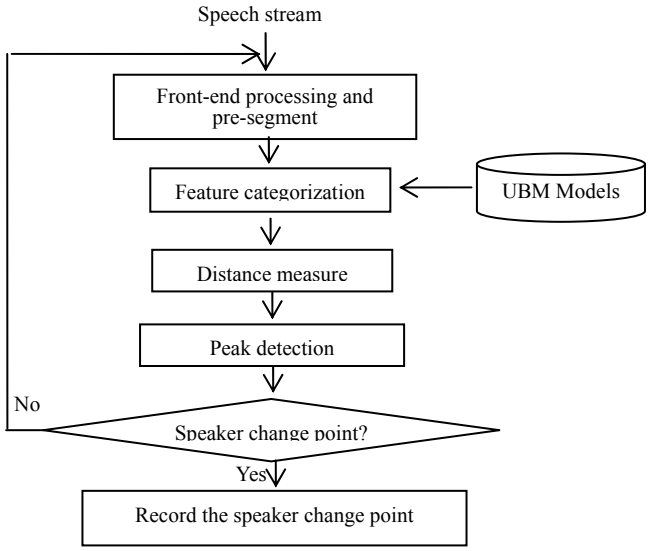

Figure 1: A brief flow diagram for speaker change detection

where $D(i, j)$ is the distance between *i-th* window and *j-th* window, $D^{i}_{\min,left}$ is the left minima around the peak, and θ_i and Th_i are dynamic thresholds. Fig. 3 shows the constraints to find a speaker boundary.

The first two conditions in Eq. (1) guarantee a local peak exists. Because of the constraint of real-time, the second condition ensures the current distance measure is just larger than the immediate later distance, rather than the right minima around the peak. The last condition prevents very low peaks from being detected. But the thresholds are difficult to set *a priori*. If the thresholds are too small, false detection would be many; otherwise, some positive speaker change boundaries would be missed. The thresholds are affected by many factors, such as insufficient estimate data and different environment conditions. For example, the distance between adjacent windows will increase if the two segments of speech are in different environments. To obtain optimal result, an automatic threshold setting method is implemented as following:

$$Th_i = \alpha_1 \cdot \frac{1}{M} \sum_{m=1}^{M} D(i-m-1, i-m), \tag{2}$$

$$\theta_i = \alpha_2 \cdot \frac{1}{M} \sum_{m=1}^{M} D^{i-m}_{left,\min}, \tag{3}$$

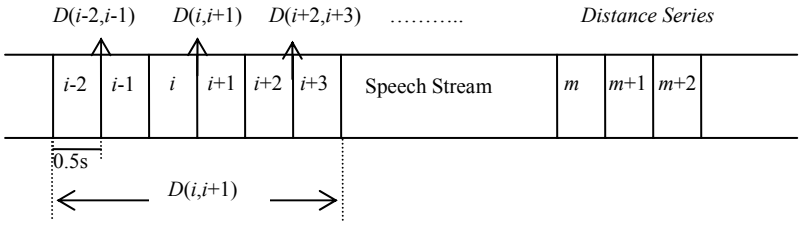

Figure 2. Illustration of speaker detection

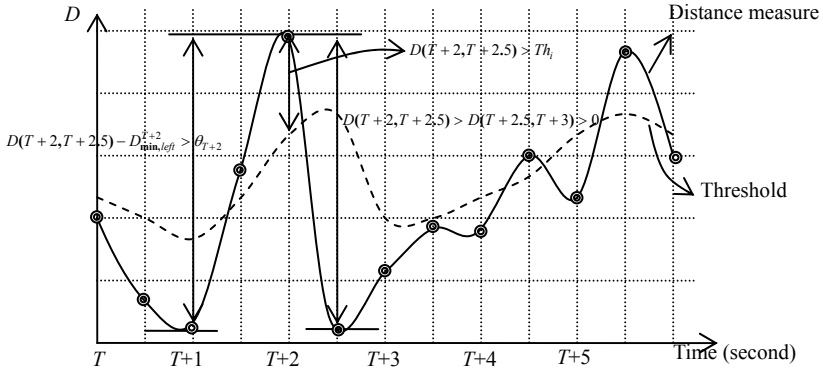

Figure 3. Illustration of finding local maximum. Speaker changes occur at (*T*+2) and (*T*+5.5).

where *M* is the number of previous distances used for predicting threshold, α_1 and α_2 are amplifiers, and are pre-defined to 1.8 and 1.2 respectively in our real-time implementation. Thus, the dynamic thresholds are set by the previous *M* successive distance. These dynamic thresholds can fit the change of environment conditions.

3 Feature Categorization and Distance Measure

In this section, we will describe the approaches of feature categorization and distance measure. We first briefly review the common distance measure used in our former speaker segmentation system and introduce the improved approach proposed by Beigi [6]. Then after some weaknesses in above two approaches are discussed, we present the alternative GMM-UBM to overcome those weaknesses. At last, we also present three distance measures based on GMM-UBM in terms of whether the effect of background is considered.

3.1 Basic Distance Measure

In our former work [8], we used Kullback-Leibler (K-L) distance to measure the distance between each two neighboring windows.

$$d = \frac{1}{2} tr[(\boldsymbol{C}_i - \boldsymbol{C}_j)(\boldsymbol{C}_j^{-1} - \boldsymbol{C}_i^{-1})] + \frac{1}{2} tr[(\boldsymbol{C}_j^{-1} + \boldsymbol{C}_i^{-1})(\boldsymbol{u}_i - \boldsymbol{u}_j)(\boldsymbol{u}_i - \boldsymbol{u}_j)^T] \quad (4)$$

The distance is composed of two parts. The first part is determined by the covariance of two segments and the second is determined by covariance and mean. Since the mean is easily biased by different environment condition, we will not consider the second part and only the first part is used to represent the distance, based on the work [1]. It is also similar to the Cepstral Mean Subtraction (*CMS*) method to compensate the effect of environment or channel mismatch. The final distance is called *divergence shape distance*, which is defined by,

$$d = \frac{1}{2} tr[(\boldsymbol{C}_i - \boldsymbol{C}_j)(\boldsymbol{C}_j^{-1} - \boldsymbol{C}_i^{-1})] \quad (5)$$

Suppose two speech clips are said by the same speaker, the distance between them would be small; otherwise, the distance would be large. So, here is a simple criterion: if the distance between two speech segments is larger than a threshold, these two segments could be considered as being said by different speakers.

Although this basic distance measure can tell the dissimilarity of sets of features, it assumes all features are related to speech, and can not distinguish the speech part and non-speech part. However, non-speech features or background noise features often exist in some speech segments, and they will disturb the measurement between speakers because those features are also considered as speech part. To solve such problem, Beigi [6] applied the K-Means algorithm to categorize the features into different types in order to improve this simple distance measure-based approach.

3.2 K-Means algorithm for feature categorization and distance measure

In Beigi's work [6], in order to discriminate different conditions, which include the difference between speakers, channels and environments, he hypothesized that there are three different types of features in every window: the silence or background-related features which are almost similar in both adjacent windows, the speech-related features which have no contributions to discriminate the different speakers and the speaker-related features which could truly tell the diversity of different speakers.

Therefore, K-Means algorithm is implemented to achieve three clusters for each window. The proposed distance to measure the dissimilarity between two adjacent windows is:

$$D = \frac{d_{ave} \max(d_{i,j})}{d_g \min(d_{i,j})}, \quad (6)$$

where $d_{i,j}$ represents the distance between clusters i $(0 \le i < 3)$ in former window and cluster j $(0 \le j < 3)$ in succeeding window; d_{ave} is the average of all $d_{i,j}$; and d_g is the distance between the two adjacent windows with a single Gaussian fitting to each window. He assumed that the maximum of $d_{i,j}$ is the features which can tell the diversity of speakers, and the minimum of $d_{i,j}$ represents the distance between two similar backgrounds. All d_* are calculated from K-L distance.

In this approach, max($d_{i,j}$) and min($d_{i,j}$) are supposed to be related with the distance between speaker features and the distance between background features, individually. However, it is almost impossible to discriminate background-related, speech-related and speaker-related features by simple clustering. In the case of the background changes, the maximum distance and minimum distance of $d_{i,j}$ might not represent the speaker's information and background information respectively, which makes the distance measure inaccurate. It will be very helpful for distance measure if we can know the exact type of each cluster. Thus, GMM-UBM is used to solve these problems.

3.3 GMM-UBM for feature categorization and its improved distance measures

GMM-UBM is an off-line trained GMM model using a mass of training data in order to represent the characteristics of these data. In many speaker recognition applications, the training data set is not enough to model the speaker's characteristics. Thus, those speaker models can be derived from adapting the parameters of UBM using the small training data by *maximum a posteriori* (MAP). Another use of UBM is to select more reliable channels for speech and speaker recognition if the plenty of data from each channel are pre-trained to a UBM separately. For example, suppose to create the UBMs for each kind of channels and backgrounds in advance, the credible channel will be selected according to MAP principle in practical applications. It is helpful to employ the corresponding tools when channel and/or background are selected.

In our approach, speaker independent UBM is trained off-line by using plenty of speech data in TV broadcasting news through EM algorithm. Such a UBM model represents the global speaker characteristics. Thus, for each feature vector in the speech stream, we can get its confidence of relating to speaker based on the UBM model.

Let the trained GMM-UBM be denoted as $G(\omega_s, \mathbf{m}_s, \Sigma_s)$ $(0 < s \le S-1)$, where S is the number of Gaussians in GMM, ω_s, $\boldsymbol{m}_s$ and Σ_s are the weight, mean and deviation of each Gaussian component respectively; and the feature vector of the *i*th window in speech stream is $\boldsymbol{X}_i = (\boldsymbol{x}_0^i, \boldsymbol{x}_1^i, ..., \boldsymbol{x}_{k_i-1}^i)$, where k_i is the number of frames in *i*th window. Thus, the confidence can be represented by the likelihood

probability function, which is defined by:

$$p_l^i(\boldsymbol{x}_l^i \mid G) = \sum_{s=1}^{S} \omega_s p_s(\boldsymbol{x}_l^i), \tag{7}$$

where

$$p_s(x_l^i) = \frac{1}{(2\pi)^{N/2} \mid \sum_s \mid^{1/2}} \exp\left\{ -\frac{(x_l^i - m_s)^T \sum_s^{-1} (x_l^i - m_s)}{2} \right\}. \tag{8}$$

It can be assumed that the features whose likelihood probabilities are relatively high have a high confidence to represent the speaker's characteristics, and the features whose likelihood probabilities are low are the unreliable speech or silence. Furthermore, we also assume that unreliable speech brings more non-speaker information than reliable speech. According to these assumptions, we also classify the feature vectors in a window into three clusters: reliable speech, doubtful speech and unreliable speech, which is similar to the three clusters in [6] which are related to speaker, speech and background noise, respectively. In our real implementation, we select the feature vectors which are in the top one third confidences as reliable speech, the feature vectors in the least one third as unreliable speech, and the feature vectors are in the middle one third as doubtful speech. It is simple, but it works well.

Feature categorization base on UBM is a little different with traditional audio classification [10] [11]. Audio classification is usually to classify the audio segment into speech, music, silence, etc. Generally, it classifies the whole audio clip to the dominant audio type, often by a *hard decision*. However, even for a segment which is classified as speech, it still may contain some speech and noise frames synchronously. It will be better if we can discriminate them. It will be also helpful if we can discriminate the strong speaker-related frames and weak speaker-related frames. Thus, we use GMM-UBM to give a confidence to each frame and make a *soft decision*.

Similar to the Eq. (6) proposed by Beigi [6], we can define the distance correspondingly, considering the cluster of reliable speech and unreliable speech.

$$D_1 = \frac{d_{ave} d_{rs}}{d_g d_{us}}, \tag{9}$$

where d_{rs} and d_{us} are computed by the corresponding reliable speech parts and unreliable speech parts between two adjacent windows respectively; and all d_* can defined as Eq. (5). In this equation, it can be seen that the distance between two reliable speech segments is corresponding to maximum distance in K-Means algorithm, and the distance between two unreliable speech segments is corresponding to minimum distance.

Moreover, in order to eliminate the influence of doubtful speech, we can normalize the reliable speech part through divided by the unreliable part. Thus, after ignored d_{ave} and d_{g}, the Eq. (9) can be modified:

$$D_2 = \frac{d_{rs}}{d_{us}}. \tag{10}$$

Furthermore, since focusing on the change of speaker identity, not caring about whether the background conditions change or not, we can even ignore the effect of background and only consider the most reliable speaker-related frames. Thus, Eq. (10) can be modified:

$$D_3 = d_{rs} . \tag{11}$$

Eq. (11) means only reliable speech component itself in each window is concerned and other information of environment or background is discarded.

4 Experiment results

In this section, database information is described first. Then we will compare the performance of feature categorization between K-Means and UBM. The comparison on different distance measures is also presented. Finally, the real-time speaker segmentation is performed.

4.1 Database

The evaluation of the proposed speaker change detection is performed on Hub-4 1997 English Broadcast News Speech Database. The database consists of about 97 hours news broadcasting, which are from different radios, such as CNN, ABC, CRI and C-SPAN. About 10 hours speech data is selected randomly for training speaker independent GMM-UBM, and the remaining speech data is for testing. In this database, each file is about 30 minutes or 60 minutes, and there are about 30 speakers and about 60-80 speaker changes in each file.

This database is originally for Spoken Document Retrieval. We use it in our experiment since it is more suitable for our intended application. The ground truth can be got from its accompanying transcripts.

We use 3 second speech data as our basic segmentation unit. This unit size has been determined from the statistics of our experiments. This size is critical since if it is too short, insufficient data will not provide enough information to discriminate the diversity between speakers; otherwise, if it is too long, the missed detection will occur more frequently.

Therefore, we should analyze the distribution of length of speaker segments. Fig. 4 shows a histogram for the length of speaker segments in training database. It illustrates that there are about 5% speaker segments are less than or equal to 2 seconds, and 10% less than 3 second. We tested the performance with the window being 2 seconds or 3 seconds, it was observed that the performance decreased dramatically when window is 2 second. Thus, we selected 3 seconds as a window unit size. That is to say, for those speaker segments which are less than 3 second, the segmentation results are not reliable.

4.2 Training GMM-UBM

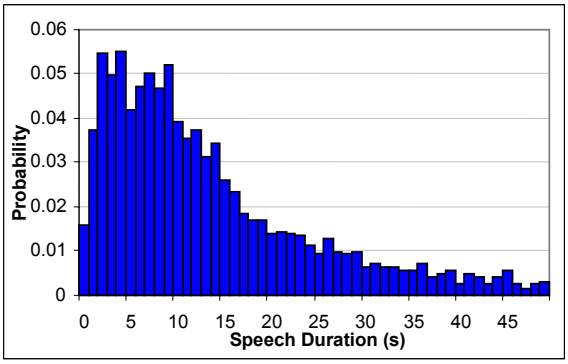

Figure 4. The histogram for the length of speaker segment

Twenty broadcasting news files are randomly selected to train speaker independent GMM-UBM. Speech data is extracted according to ground truth files. Furthermore, the silence segments in speech data are discarded using simple energy threshold so that only speech data is considered. The clean training speech data is about 10 hours. These data are blocked into 25ms-frame without overlapping. 10-order LSP and 16-order MFCC are extracted from each frame. We pool all features to train two 64-Gaussians UBMs: LSP UBM and MFCC UBM, by EM algorithm. Both UBMs are gender-independent.

4.3 Experiments on feature categorizing

False alarm rate (FAR) and missed detection rate (MDR) are used to compare the performance of segmentation when different feature categorization approaches are employed. In speaker segmentation, FAR is calculated as:

$$FAR = \frac{number\ \ of\ \ false\ \ detection}{number\ \ of\ \ fasle\ \ detection + number\ \ of\ \ true\ \ speaker\ \ change} \times 100\%, \tag{12}$$

and MDR is calculated as:

$$MDR = \frac{number\ \ of\ \ miss\ \ detection}{number\ \ of\ \ true\ \ speaker\ \ change} \times 100\%. \tag{13}$$

Three feature categorization approaches are compared in our speaker change detection system. The first one is without feature categorization. That is, we pool all feature vectors without feature vectors categorizing to compute the distance. The second is feature categorization based on K-Means. The third is feature categorization based on UBM. Fig. 5 and Fig. 6 show the comparative ROC results among these three approaches when using LSP and MFCC respectively. The three

curves in each figure are denoted as “original”, “K-Means” and “UBM” respectively. which represent different feature categorization approach. For distance measure, “original” uses the basic distance measure; while “K-Means” and “UBM”, uses Eq. (6) and Eq. (9). The thresholds are fixed in this experiment.

From these two figures show, it can be seen that if whichever feature is employed, when the same MDR error is allowed, the false alarm rate base on UBM is least. In more detail, it is,

$$FAR_{UBM} < FAR_{K\text{-}Means} < FAR_{original}. \quad (14)$$

For example, when MDR is 30%, the FAR of UBM is 5% less than K-Means and about 11% less than original system when LSP is used.

Therefore, UBM for feature categorization has better performance than speaker segmentation without feature categorization or with feature categorization based on K-Means. This is because UBM can supervise the reliable speech part and unreliable speech part in experiments. Thus it can focus on the change of speaker characteristics.

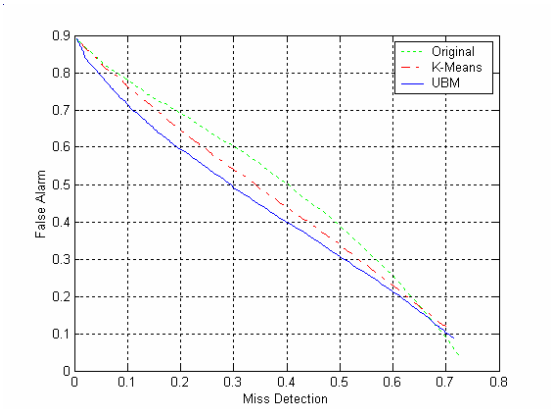

Figure 5. The ROC curves of three approaches with LSP feature

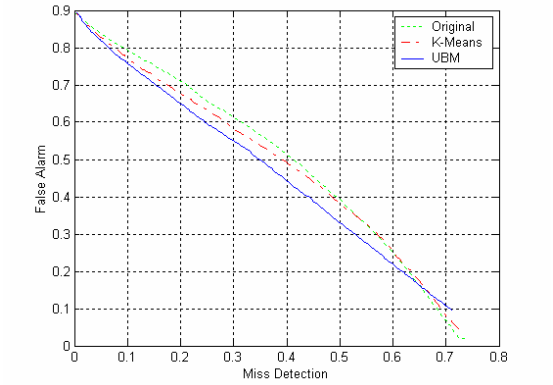

Figure 6. The ROC curves of three approaches with MFCC feature

4.4 Comparison on different distance measures

After feature categorization based on GMM-UBM, three distances are compared as defined in Eq. (9), (10) and (11). In this experiment, we compared the segmentation performance when different distance measures are employed.

Fig. 7 and Fig. 8 illustrate the ROC curves with different distance measures of Eq. (9), (10) and (11), which are named D_1, D_2 and D_3, employing LSP and MFCC, respectively. In Fig.7, where LSP is employed, when MDR is 30%, the FAR of D_3 is 15% less than D_1 and about 21% less than D_2.

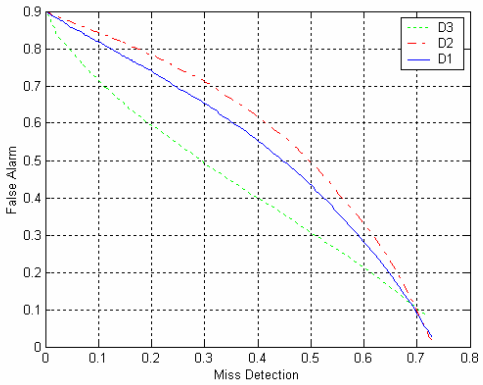

Figure 7. The ROC curves using different distance measure with LSP

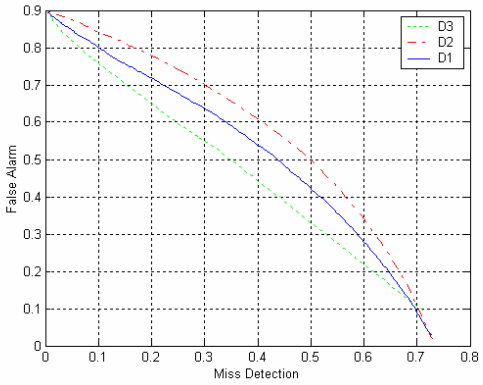

Figure 8. The ROC curves using different distance measure with MFCC

From Fig. 7 and Fig. 8, we can see, D_3, which is not considered doubtful speech and unreliable speech shows the best performance. The possible reason is that what we care is not to detect environment conditions but the change of speakers, while Beigi concerned not only the speaker change, but also the change of environment channel [6]. Thus, we should grasp the change of speaker

characteristics, and ignore the influence of background in real-time speaker segmentation. On the contrary, both D_1 and D_2 employ the unreliable speech and doubtful speech. This consideration may import the information channel or environment and cause FAR increase. The figures also illustrate that the performance of D_1 is better than D_2's. A possible reason is that doubtful speech, which is employed in D_1, but not in D_2, is intervenient between reliable speech and unreliable speech. Thus, doubtful speech may also bring some speech information, which can counteract some part of information of channel or environment brought by unreliable speech.

4.5 Real-time Speaker segmentation

LSP and MFCC are both the reliable features in speaker recognition, and they show the different aspects about speaker characteristics. In our experiment, LSP and MFCC are fused simply to get a more robust distance, which is defined by:

$$D_{final} = 0.5 \times D_{LSP} + 0.5 \times D_{MFCC}, \quad (15)$$

where D_{LSP} and D_{MFCC} are defined as Eq. (11).

Fig. 9 shows an example of real-time speaker segmentation on about 100-second-long speech. The solid line in this figure represents the true speaker changes, and the dot line represents the final distance measure.

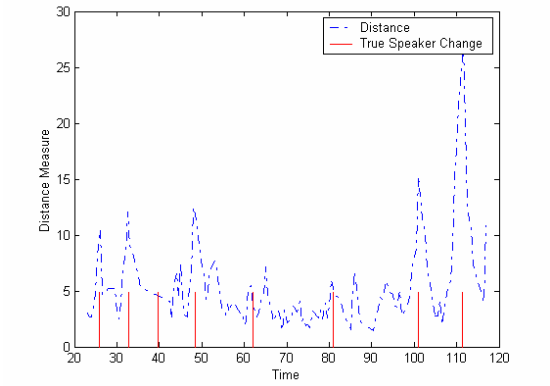

Figure 9. Real-time Speaker segmentation based on distance measure

In this experiment, we use the dynamic thresholds described to detection the potential speaker change boundaries. The final result of real-time speaker segmentation is MDR=24.5%, FAR=36.4%.

4.6 Discussion

From the above experiments, it can be seen that UBM is a promising approach for feature categorization and the robust discriminative distance measure in

real-time speaker segmentation. UBM can distinguish the reliable speech part and unreliable speech part. It makes speaker segmentation more robust comparing the other two approaches. Moreover, the feature categorization based on UBM is low computation complex so that it can satisfy real-time requirement.

However, the MDR and FAR is still a little high in speaker segmentation system. This is mostly because the following reasons. First, short segments which are shorter than 3 second result in MDR greatly. Because the basic unit of our algorithm is 3 second, speaker changes may be not detected if the length of speech is less than 3 second. These short segments will cause 5%-10% MDR. Second, although only the reliable speech part selected by UBM is considered, it may still bring the noise information, which will increase FAR. Third, with the constraint of real-time, segmentation results can not be refined by iterative operation, and can not use the global data. This difficulty does not exist in the off-line system.

5 Conclusion

In this paper, we describe the procedure of speaker change detection and present the GMM-UBM algorithm for speaker segmentation. Due to the non-speech frames in speech stream, UBM is presented to categorize features into three parts. Experiments showed feature categorization based on GMM-UBM is helpful to real-time speaker segmentation. Experiment also showed that the distance measure which only used reliable speech frames got a much better results.

6 Reference

[1] J. P. Campbell, JR. Speaker Recognition: A Tutorial, *Proc. of the IEEE*, vl.85, No. 9 (1997), pp.1437-1462.

[2] S. Chen and Gopalakrishman, Speaker, Environment and Channel Change Detection and Clustering via the Bayesian Information Criterion, *Proc. Of DARPA Broadcast News Transcription and Understanding Workshop* (1998).

[3] L. Wilcox, F. Chen, D. Kumber, and V. Balasubramanian, Segmentation of Speech Using Speaker Identification, *Proc. Of IEEE International Conference on Acoustics, Speech, and Signal Processing* (1994), pp.161-164.

[4] K. Mori and S. Nakagawa, Speaker Change Detection and Speaker Clustering Using VQ Distortion for Broadcast news Speech Recognition, *Proc. Of IEEE International Conference on Acoustics, Speech, and Signal Processing* (2001), pp.413-416.

[5] D. A. Reynolds, T. F. Quatieri and R. B. Dunn, Speaker Verification Using Adapted Gaussian Mixture Models, *Digital Signal Processing* 10 (2000), pp.19-41.

[6] H. S. Beigi, Maes, S., Speaker, Channel and Environment Change Detection, In: *World Congress of Automation* (1998).

[7] O. Pietquin, L. Couvreur, P. Couvreur, Applied Clustering for Automatic Speaker-based segmentation of Audio Material, *Belgian Journal of Operations Research, Statistics and Computer Science (JORBEL)* (2002).
[8] L. Lu, H. J. Zhang, Speaker Change Detection and tracking in Real-time News Broadcasting Analysis, accepted by *ACM Multimedia* (2002)
[9] P. Delacourt, C.J. Wellekens, DISTBIC: A speaker-based segmentation for audio data indexing, *Speech Communication* 32 (2000), pp.111-126.
[10] L. Lu, H. Jiang and H. J. Zhang, A Robust Audio Classification and Segmentation Method, *Proceeding of 9th ACM Multimedia* (2001), pp.203-211.
[11] L. Lu, S.Z. Li and H. J. Zhang, Content-based Audio Segmentation Using Support Vector Machines, *Proceeding of ICME01* (2001), pp.956-959.