EFFECTS OF BACKGROUND DATA DURATION ON SPEAKER VERIFICATION PERFORMANCE

Cemal HANİLÇİ [*]
Figen ERTAŞ[*]

Abstract: Gaussian mixture models with universal background model (GMM-UBM) and vector quantization with universal background model (VQ-UBM) are the two well-known classifiers used for speaker verification. Generally, UBM is trained with many hours of speech from a large pool of different speakers. In this study, we analyze the effect of data duration used to train UBM on text-independent speaker verification performance using GMM-UBM and VQ-UBM modeling techniques. Experiments carried out NIST 2002 speaker recognition evaluation (SRE) corpus show that background data duration to train UBM has small impact on recognition performance for GMM-UBM and VQ-UBM classifiers.

Keywords: Speaker verification, Gaussian mixture model, Vector Quantization, Universal background model

Arkaplan Veri Süresinin Konuşmacı Doğrulama Performansına Etkisi

Özet: Gauss karışım modeli genel arka plan modeli (GKM-GAM) ve vektör nicemleme genel arka plan modeli (VN-GAM) konuşmacı doğrulamada sık kullanılan iki yöntemdir. Genellikle GAM modeli fazla sayıda farklı konuşmacının bulunduğu bir kümeden seçilen saatlerce uzunluktaki ses işaretleri kullanılarak eğitilir. Bu çalışmada, GAM modelinin eğitiminde kullanılan veri miktarının metinden bağımsız konuşmacı doğrulama performansına etkisi incelenmektedir. NIST 2002 konuşmacı tanıma değerlendirme veritabanı ile GKM-GAM ve VN-GAM yöntemleri kullanılarak yapılan deneysel çalışmalar arka plan modelini eğitmek için kullanılan veri miktarının konuşmacı tanıma performansına çok fazla etkisinin olmadığı görülmüştür.

Anahtar Kelimeler: Konuşmacı doğrulama, Gauss karışım modeli, Vektör nicemleme, Genel arka plan modeli

1. INTRODUCTION

Speaker recognition aims to identify speaker from a given speech signal which refers to two different tasks: *speaker identification* and *speaker verification*. The goal of speaker identification is to determine which one of a known speakers best matches with the input speech signal spoken by unknown speaker. This is known as *closed-set* speaker identification since recognizer tries to match input speech to the one speaker known in database. In speaker verification, the goal is to determine whether speaker is who he or she claims to be. This is referred to as *open-set* verification (Kinnunen and Li, 2011).

[*] Uludağ University, Faculty of Engineering and Architecture, Dept. of Electronic Engineering, Gorukle 16059, Bursa.
İletişim Yazarı: C. Hanilçi (chanilci@uludag.edu.tr)

The standard speaker recognition system consists of two phases: training and recognition. In training, given a training speech sample of a particular known speaker, the features are extracted and then a speaker model is trained using training feature vectors. In recognition step, the features are extracted from unknown speaker's test speech sample and a match score is computed using each speaker model in the database and the speaker model which produces the maximum score determined as the identity of unknown speaker (speaker identification) or if the similarity score between the feature vectors and claimed speaker model is above a threshold the unknown speaker is accepted (speaker verification).

In recent studies, generally long speech samples (around 5 minutes long) have been used for speaker verification. The most popular speaker recognition corpora NIST speaker recognition evaluations (SRE) provide long speech utterances as the core-task. This is probably due to the fact that features extracted from a long speech sample captures speaker characteristic more than features extracted from short utterances. However, in real-time applications this is problematical. In user-convenient applications it is more suitable to ask speaker to produce short speech samples. There exists a number of studies which consider speaker recognition using short utterances and as expected the recognition performance reduces when short speech samples are used.

Current state-of-the-art speaker recognition systems use universal background model (UBM) approach. Since speaker verification is a two-class pattern recognition problem one class represents the speaker and the other class represents the alternative speakers. The UBM approach models the alternative class which will be described in the next section in detail. UBM method has become an integral part of classifiers for speaker verification. Gaussian mixture model (GMM) (Reynalds et. al., 2000), vector quantization (VQ) (Hautamaki et. al., 2008), GMM supervector (GMM-SVM) (Campbell et. al., 2006), joint factor analysis (JFA) (Kenny et. al. 2007) and i-vector (Dehak et. al., 2011) systems all use the UBM method, initially. Thus training UBM model is one of the most important part of designing a classifier. Generally UBM model is trained using large number of speech samples (mostly a few thousands of speech samples) from several speakers which do not exist in training speaker set. However, there is no objective measurement to determine the amount of speech samples to train UBM. It is believed that the more data used to train UBM, the better recognition performance. In this paper, we experimentally analyze the effect of data duration to train UBM on speaker verification performance for GMM-UBM and VQ-UBM classifiers.

2. LIKELIHOOD RATIO DETECTOR FOR SPEAKER VERIFICATION

Given a speech sample X speaker verification is a hypothesis test of two possible hypotheses:

$$H0\colon X \text{ was produced by claimed speaker } S$$
$$H1\colon X \text{ was not produced by claimed speaker } S$$

and the verification system tries to decide which of these two hypotheses is true. When the likelihoods of both hypotheses are known, the optimum decision is made by the likelihood ratio which is defined as:

$$\frac{p(X|H_0)}{p(X|H_1)}\begin{cases} > \theta & Accept\ H_0 \\ \leq \theta & Accept\ H_1 \end{cases} \tag{1}$$

where $p(X|H_0)$ and $p(X|H_1)$ are the likelihoods of the hypotheses H_0 and H_1, respectively (Reynolds et. al., 2000). θ is the decision threshold to accept or reject the hyphothesis H_0.

Generally, the logarithm of the likelihood ratio is used which gives the log-likelihood ratio (Figure 1)

$$\Lambda(X) = \log p(X|H_0) - \log p(X|H_1) \tag{2}$$

Often, H_0 and H_1 are represented by models denoted λ_{hyp} and $\lambda_{\overline{\text{hyp}}}$ where they characterize the hypothesized speaker S and the alternative hypothesis H_1, respectively. The model λ_{hyp} is well defined and estimated using the feature vectors extracted from the training speech sample of speaker S. However, $\lambda_{\overline{\text{hyp}}}$ is problematical since it requires to represent the entire space of possible alternative speakers except the hypothesized speaker *S*. The $\lambda_{\overline{\text{hyp}}}$ model is generally estimated by pooling a large number of speech samples from several speakers. This is known as *universal background model* (UBM) and is denoted by λ_{UBM} (Reynolds et. al., 2000). UBM is a large model trained for speaker-independent representation of feature space. However, there is no theoretical measure to determine the optimum amount of data used to estimate UBM model.

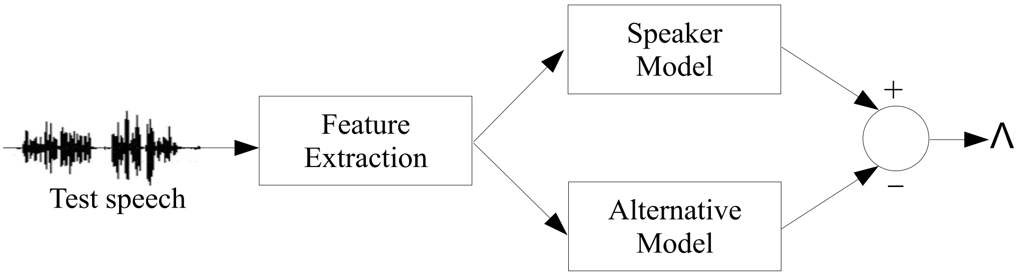


Figure 1:
Likelihood Ratio Based Speaker Recognition

3. UBM BASED RECOGNITION SYSTEMS

Figure 2 shows the general steps of UBM based speaker recognition system. The first step of recognition system is pooling large number of speech samples from several speakers and then extracting features from these speech samples to train UBM model. When a new speaker is enrolled into the system, first the features are extracted from speaker's training speech sample and then a speaker model is adapted using training feature vectors and UBM model via *maximum a Posteriori* (*MAP*) adaptation. In the recognition step, feature vectors are extracted from the unknown sample and compared with the model of claimed speaker and UBM model and a similarity score is computed as described in the previous section. If the similarity score is above the threshold the unknown speaker is accepted or rejected otherwise.

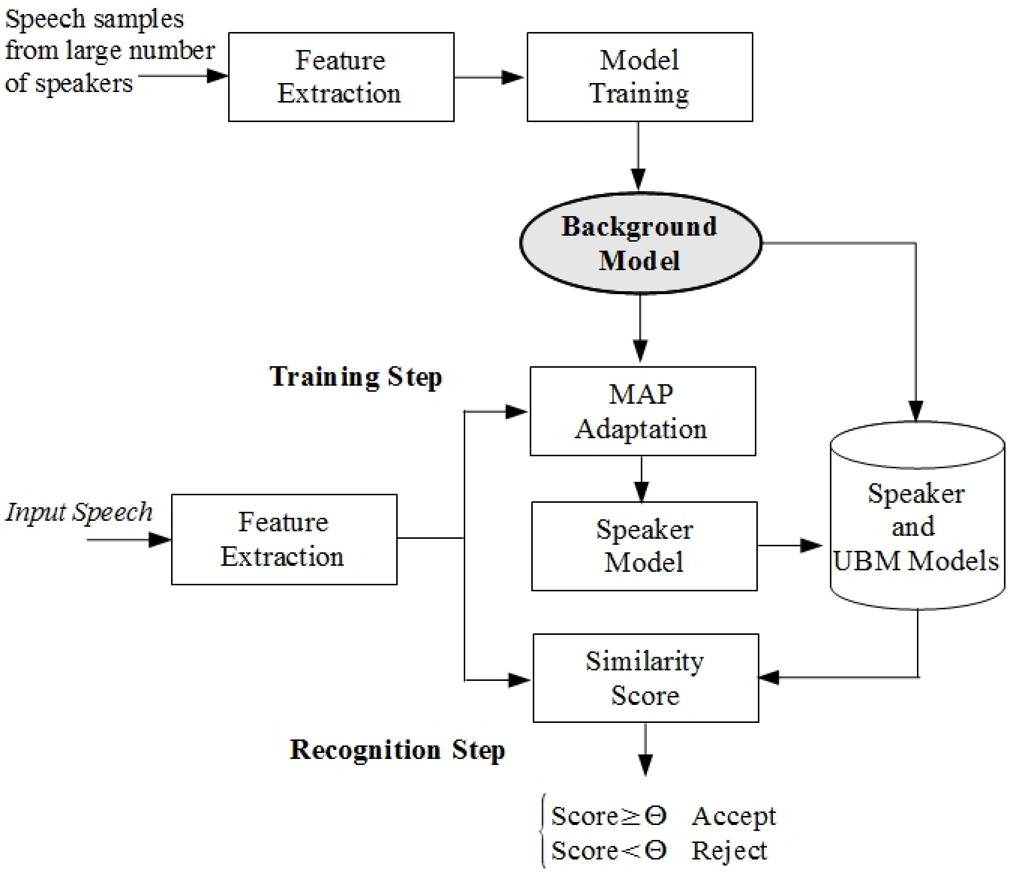


Figure 2:
General speaker recognition system

Gaussian mixture model (GMM) is one of the most popular UBM based modeling technique. In GMM, likelihood function is defined as a weighted linear combination of *M* multivariate Gaussian densities

$$p(\mathbf{x}|\lambda) = \sum_{i=1}^{M} w_i p_i(\mathbf{x}) \tag{3}$$

where w_i is the mixture weights constrained as $\sum_{i=1}^{M} w_i = 1$ and $p_i(\mathbf{x})$ is a *D*-variate Gaussian density with the mean vector $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_i$

$$p_i(\mathbf{x}) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\} \tag{4}$$

Training a UBM model in GMM, consists of estimating the model parameters $\lambda_{\text{UBM}} = \{w_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}_{i=1}^{M}$ using pooled background feature vectors $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$ extracted from speech samples from several speakers by maximizing the objective function. The average log-likelihood is generally used as objective function. The average log-likelihood of **X** given a GMM model λ is defined as

$$LL_{\text{avg}}(X|\lambda) = \frac{1}{T}\sum_{t=1}^{T}\log\sum_{i=1}^{M} w_i p_i(\mathbf{x}_t|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \tag{5}$$

the popular expectation maximization (EM) algorithm is used to maximize the log-likelihood for a given training data **X**. To derive a speaker model from the UBM model λ_{UBM} , MAP adaptation is used after UBM training. In early studies, it was shown that adapting only mean vectors gives better recognition performance than adapting all parameters in UBM (weights, mean and covariances). Given speaker training feature vectors $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T\}$ and UBM model $\lambda_{UBM} = \{w_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}_{i=1}^{M}$, the MAP adapted mean vectors of the ith mixture component is given by

$$\widehat{\boldsymbol{\mu}}_i = \alpha_i E_i(\mathbf{x}) + (1-\alpha_i)\boldsymbol{\mu}_i \tag{6}$$

where

$$\alpha_i = \frac{n_i}{n_i + r} \tag{7}$$

$$E_i(\mathbf{x}) = \frac{1}{n_i}\sum_{t=1}^{T}\Pr(i|\mathbf{x}_t)\,\mathbf{x}_t \tag{8}$$

$$\Pr(i|\mathbf{x}_t) = \frac{w_i p(\mathbf{x}_t)}{\sum_{j=1}^{M} w_j p_j(\mathbf{x}_t)} \tag{9}$$

where $\boldsymbol{\mu}_i$ is the mean vector of ith Gaussian component in the UBM model, $\widehat{\boldsymbol{\mu}}_i$ is the MAP adapted mean vector of the speaker model and *r* is the relevance factor. The same mixture weights, w_i and covariance matrices Σ_i are used in both λ_{SPK} and λ_{UBM} (Reynolds et. al., 2000, Kinnunen and Li, 2011). For more details about GMM and GMM-UBM methods readers are refered to Reynolds et. al (2000).

In the recognition step, unknown feature vectors $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\}$ and claimed speaker and UBM models λ_{SPK} and λ_{UBM} are used to compute log-likelihood ratio score.

Vector quantization (VQ) in turn, another simple but powerful classifier and has successfully been used in speaker recognition (Kinnunen et. al., 2008). In VQ, given background feature vectors $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T\}$ the aim is to find a set of codevectors known as codebook which is denoted by $\mathbf{C}_{\text{UBM}} = \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_M\}$, where $M \ll T$ which minimizes a given objective function. Mean squared error (MSE) is usually used as objective function:

$$\text{MSE}(\mathbf{X}, \mathbf{C}) = \frac{1}{T}\sum_{t=1}^{T}\min_{1\le m\le M}\|\mathbf{x}_t - \mathbf{c}_m\|^2 \tag{10}$$

where $\|\mathbf{x}_t - \mathbf{c}_m\|^2$ is the squared Euclidean distance between the vectors $\mathbf{x}_t$ and $\mathbf{c}_m$. Standard *K*-means algorithm is used to train C_{UBM}. After UBM model is trained, the speaker model C_{SPK} (speaker codebook) is adapted from UBM model C_{UBM} using speaker training feature vectors $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T\}$ via MAP adaptation (Hautamaki et. al. 2008, Kinnunen et. al., 2009, Hanilci and Ertas, 2011).

$$\hat{\mathbf{c}}_m = w_m \bar{\mathbf{x}}_m + (1 - w_m)\mathbf{c}_m \tag{11}$$

where

$$w_m = \frac{|S_m|}{|S_m| + r} \tag{12}$$

where $|S_m|$ is the number of training vectors assigned to the mth codebook by minimum distance criterion, $\bar{\mathbf{x}}_m$ is the mean vector of these vectors and *r* is the relevance factor.

Given a sequence of feature vectors, $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\}$, and the claimed speaker model, , we compute the log-likelihood ratio and compare it against a threshold to make the decision. In the VQ model, the log-likelihood is related to the negative square error given in (10). Thus, the match score can be de fined as

$$\Lambda = -\mathrm{MSE}(Y, C_{\mathrm{SPK}}) - \big(-\mathrm{MSE}(Y, C_{\mathrm{UBM}})\big) = \mathrm{MSE}(Y, C_{\mathrm{SPK}}) - \mathrm{MSE}(Y, C_{\mathrm{UBM}}) \tag{13}$$

4. EXPERIMENTAL SETUP

4.1. Corpus, Classifier Design, Features and Performance Criteria

Speaker recognition experiments are carried out on the NIST 2002 SRE corpus (NIST, 2002) which consists of conversational telephone speech sampled at 8 kHz and transmitted over different cellular networks. It involves 330 target speakers (139 males and 191females) and 39259 verification trials (2982 targets and 36277 impostors). For each target speaker, approximately two minutes of training data is available whereas duration of the test utterances varies between 15 seconds and 45 seconds. GMM-UBM and VQ-UBM are used as the classifier. Two gender-dependent background models with the model order 512 (number of Gaussians in GMM-UBM and number of code vectors in VQ-UBM) are trained using the NIST 2001 SRE corpus (NIST, 2001). There are 144 male and 149 female speakers in the NIST 2001 SRE database to train UBM data. We used background data duration varies from 5 minutes to 110 minutes for each gender. The background data is selected from active speech portions (after non-speech frames are dropped). Non-speech frames are dropped using adaptive energy based voice activity detection (VAD) (Kinnunen et. al., 2009). Energy VAD measures the frame energy by calculating standard deviation of the frame and compares it to the threshold. Standard deviation of a frame is calculated by

$$E_i = 20\log\left(\frac{1}{N-1}\sum_{j=1}^{N-1}(s_i(j) - \hat{s}_i)^2\right)^{1/2} \tag{14}$$

Where $s_i(j)$ is the j^{th} sample of the i^{th} frame, $\hat{s}_i$ is the sample mean of the frame and *N* is the number of samples in frame s_i. The i^{th} frame is detected as speech if $E_i > (\max_{\mathrm{j}} \mathrm{E}_{\mathrm{j}} - 30)$, $E_j > -55$.

Standard mel-frequency cepstral coefficients (MFCCs) are used as the features. Each frame is multiplied by a 30 msec Hamming window, shifted by 15 msec. From the windowed speech frames, magnitude spectrum using Fast Fourier Transform (FFT) is computed and spectrum is processed through a 27-channel triangular filterbank and logarithmic filterbank outputs are converted into MFCCs using the discrete cosine transform (DCT). After RASTA filtering the 12 MFCCs, their first and second order time derivatives (Δ and ΔΔ) are appended. The last two steps are energy-based voice activity detector (VAD) followed by cepstral mean and variance normalization (CMVN).

As the performance criteria, we consider both equal error rate (EER) and minimum detection cost function (MinDCF). EER is the threshold value at which false alarm rate (P_{fa}) and miss rate (P_{miss}) are equal and MinDCF is the minimum value of a weighted cost function which is given by $0.1 \times P_{miss} + 0.99 \times P_{fa}$. Detection error tradeoff (DET) curves are also presented to show full behavior of the proposed methods.

4.2. Experimental Results

Figure 3 shows the speaker recognition performance in terms of EER and MinDCF as a function of background data duration used to train UBM models for GMM-UBM and VQ-UBM methods. Interestingly, both methods yield quite high error rate when data duration is less than 20 minutes and both GMM-UBM and VQ-UBM gives almost the same error rate. Another interesting observation that can be made from the figure is that, both methods have a knee-point at 20 minutes and when data duration is longer than 20 minutes both methods have less variations on the recognition performance when data duration is increasing. The performance of GMM-UBM and VQ-UBM are very close to each other independent from data duration in terms of EER. However, MinDCF is a decreasing function with respect to data duration. Different from the EER case, GMM-UBM outperforms VQ-UBM for all cases.

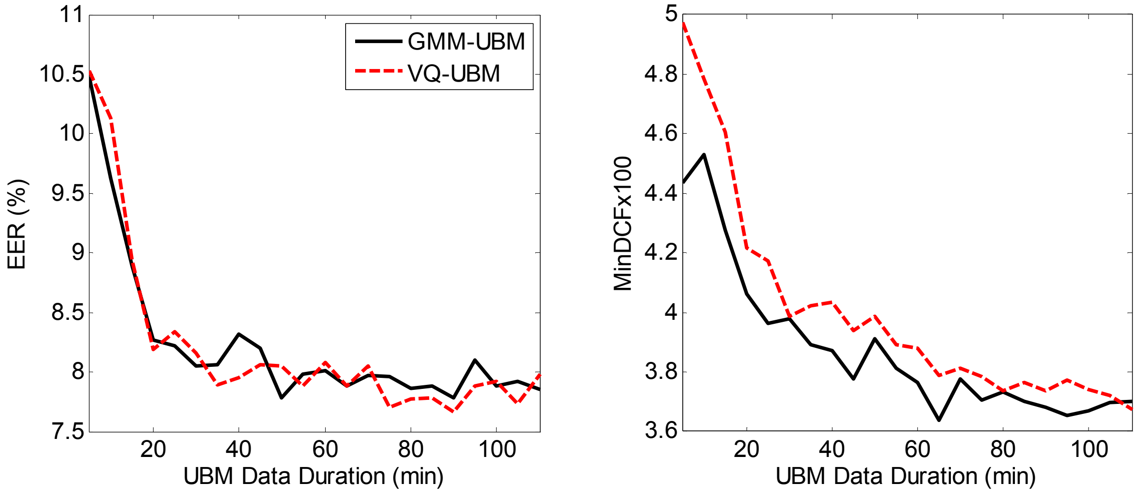
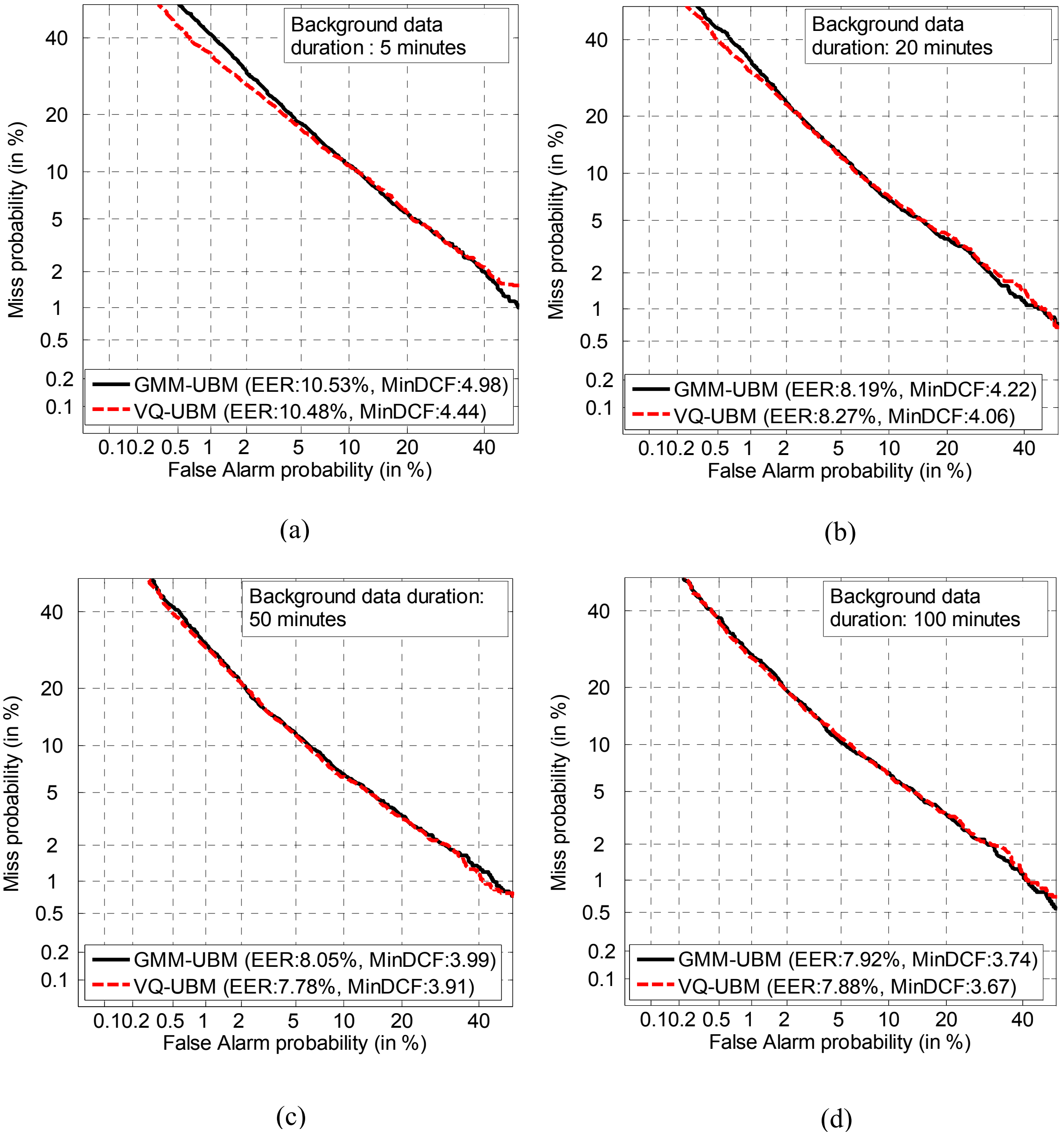


Figure 3:
EER and MinDCF values as a function of background data duration.

Figure 4 shows the DET curves for a few selected data durations to see the full behaviour of each method. There is no large difference between GMM-UBM and VQ-UBM methods for different data durations. However, for short data durations (equal or less than 20 minutes) VQ-UBM yields slightly smaller false alarm probabilities (top left corners of DET curves for Figure 4.(a) and (b)). When data duration is longer than 20 minutes this observation does not hold.

5. CONCLUSION

In this paper, we analyzed the effect of data duration to train UBM for GMM-UBM and VQ-UBM classifiers on speaker verification performance. Experimental results with NIST 2002 SRE corpus showed that small recognition accuracy was obtained when background data duration is less than 20 minutes for both GMM-UBM and VQ-UBM classifiers. However, we didn't observe so many variations on recognition rates for the values of background data duration longer than 20 minutes. In general, GMM-UBM and VQ-UBM classifiers showed similar performances and for both systems background data duration had similar effects on recognition accuracy.



(a)

(b)

(c)

(d)

Figure 4:
DET curves for different data durations

REFERENCES

1. Campbell, W., Sturim, D. E., Reynolds, D. A., Support Vector Machines Using GMM Supervectors for Speaker Verification, IEEE Signal Processing Letters, Vol. 13, No. 5, pp. 308–311, May 2006.

2. Dehak, N., Kenny, P., Dehak, R., Dumouchel, P and Ouellet, P. (2011) Front-End Factor Analysis for Speaker Verification, *IEEE Transactions on Audio, Speech and Language Processing,* 19(4), 788-798.

3. Hanilçi, C. and Ertaş, F. (2011) Comparison of the impact of some Minkowski metrics on VQ/GMM based speaker recognition, *Computers & Electrical Engineering,* 37(1), 41-56.

4. Hautamäki, V., Kinnunen, T., Kärkkäinen, I., Tuononen, M., Saastamoinen, J. and Fränti, P. (2008) Maximum a Posteriori Estimation of the Centroid Model for Speaker Verification, *IEEE Signal Processing Letters,* 15: 162--165.

5. Kenny, P., Boulianne, G., Ouellet, P. and Dumouchel, P. (2007) Joint factor analysis versus eigenchannels in speaker recognition, IEEE *Transactions on Audio, Speech and Language Processing,* 15 (4), 1435-1447.

6. Kinnunen, T., Saastamoinen, J., Hautamäki, V., Vinni, M. and Fränti, P. (2009) Comparative Evaluation of Maximum a Posteriori Vector Quantization and Gaussian Mixture Models in Speaker Verification, *Pattern Recognition Letters*, 30(4): 341--347.

7. Kinnunen, T. and Li, H. (2011) An Overview of Text-Independent Speaker Recognition: from Features to Supervectors, *Speech Communication* 52(1), 12--40.

8. NIST, (2001). http://www.itl.nist.gov/iad/mig/tests/sre/2002/index.html, Retrieved: July 2012, Subject: NIST 2002 SRE Evaluation Plan

9. NIST, (2002). http://www.itl.nist.gov/iad/mig/tests/sre/2001/index.html, Retrieved: July 2012, Subject: NIST 2001 SRE Evaluation Plan

10. Reynolds, D. A., Quatieri, T. F. and Dunn, R. B. (2000) Speaker Verification Using Adapted Gaussian Mixture Models, *Digital Signal Processing,* 10(1-3), 19-41.

Makale 01.11.2012 tarihinde alınmış, 20.12.2012 tarihinde düzeltilmiş, 21.12.2012 tarihinde kabul edilmiştir.