14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy, September 4-8, 2006, copyright by EURASIP

EFFICIENT IMPLEMENTATION OF GMM BASED SPEAKER VERIFICATION USING SORTED GAUSSIAN MIXTURE MODEL

H. R. Sadegh Mohammadi and R. Saeidi***

* Iranian Research Institute for Electrical Engineering, No. 166, Heidarkhani Ave., Narmak, Tehran, I. R. Iran
** Electrical Engineering Department, Iran University of Science and Technology, Narmak, Tehran, I. R. Iran
phone: + (9821) 77453382, fax: + (9821) 77453106, email: h.sadegh@ijece.org

ABSTRACT

In this paper a new structured Gaussian mixture model, called sorted GMM, is proposed as an efficient method to implement GMM-based speaker verification systems; such as Gaussian mixture model universal background model (GMM-UBM) scheme. The proposed method uses a sorted GMM which facilitate partial search and has lower computational complexity and less memory requirement compared to the well-known tree-structured GMM of the same model order. Experimental results show that a speaker verification system based on the proposed method outperforms that of a similar system which uses tree-structured from performance point of view. It also provides comparable performance with the GMM-UBM method despite its 3.5 times lower computational cost for a GMM of order 64.

1. INTRODUCTION

Gaussian mixture model based text-independent speaker verification has attracted the interests of many researchers in the past decade [1]. In many speaker verification applications, the accuracy and computational load are two major criteria for the selection of a proper system. Superior performance of some GMM variants compared to the other known method in this area has promoted enormous new ideas to enhance the performance and/or to reduce the computational complexity of the system. A variant of Gaussian mixture models which uses universal background model (GMM-UBM) method for speaker verification has established high performance in several NIST evaluations and has become the dominant approach in text-independent speaker verification [2]. In this method, the major computation loads are the likelihood computation for all mixtures of the UBM to select the highest scoring mixtures (top-C mixtures) and the likelihood calculation for the claimed speaker model in the system [2]. Such a system uses the majority of the processing power for scoring the Gaussian densities.

Different procedures have been reported in the literature to speed up the computation in a GMM-UBM based speaker verification system while maintaining the system error rate in an acceptable range [3], [4]. Shinoda and Lee proposed a hierarchical structure of model common to all speakers' GMMs and a multi-resolution GMM is used whose mean vectors are organized in a tree structure, with coarse-to-fine resolution when going down the tree [4]. The resulting method is known as structural background model-structural Gaussian mixture model (SBM-SGMM). To compensate the performance degradation resulted from the employment of such lower complexity methods, the application of a post-processing block, such as a neural network [5] or GMM Identifier [6] is recommended.

In this paper a new variant of GMM, called sorted GMM, is proposed as a method to reduce the computational complexity of the GMM based speaker verification systems. This method is benefited from a sorted GMM which facilitate partial search for finding the best mixture for each input target vector. The remainder of the paper is organized as follows. Section 2 presents a brief introduction to the proposed sorted GMM method. In Sections 3, the principles of the sorting GMM along with a model optimization algorithm for the proposed method are explained. Section 4 reports the implemented computer simulations and some discussions on its experimental results. The paper is concluded in Section 5.

2. SORTED GAUSSIAN MIXTURE MODEL

In GMM-UBM speaker verification, a dedicated GMM is used for each speaker. First, a single UBM is trained using a large speech corpus which contains different speech utterances from a rather large number of speakers which their speech utterances are not supposed to be evaluated by the system. Then for each target speaker a model is constructed using the UBM via Bayesian or Maximum a Posterior (MAP) adaptation method using the corresponding speaker's speech data [2]. Since the UBM and speaker models are associated to each other, a fast scoring technique can be used as follows. For each input feature vector, the top C highest scoring mixtures are determined by scoring all mixtures of the UBM (usually C is equal to 4 or 5). This needs a brute-force search. Then the claimed speaker model likelihood is calculated using only the C speaker mixtures corresponding to the top C indices selected from the UBM.

Since in the GMM-UBM method, all mixtures of a UBM are used to calculate the likelihood for each input vector, the computational complexity of the system is considerable. To increase the efficiency of the system in finding the top C mixtures, UBM's Gaussian mixtures can be ordered properly to organize a sorted structure, which hereafter will be called sorted GMM. The sorting is performed by the definition of a variable, s, which is a function of mixture's parameters. Accordingly, the top C mixtures for a given input feature vector can be found easily by intelligent use of this sorted structure. It is noteworthy that while the set of top C mixtures of the sorted GMM may not always match that of the GMM-UBM, but still it is useful for the speaker verification task. Calculation of the likelihood for the claimed speaker model is similar to what is done in GMM-UBM or SBM-SGMM. In fact, the proposal of the sorted GMM scheme is inspired by the success of a rather similar idea known as sorted codebook vector quantization in the quantization literature [7]. The next section explains the principles of the sort GMM method in a formal framework.

3. SORTED GMM PRINCIPLES

Sorted Gaussian mixture model is a new type of fast scoring GMM that is outlined here briefly in a similar fashion to what reported in [7]. Given an L -dimensional feature vector $\mathbf{x}_t = [x_{1t}, x_{2t},..., x_{Lt}]^T$ related to the speech frame at the time interval, t , and a GMM of order M , we define a *sorting parameter* $s_t = f(x_{1t}, x_{2t},..., x_{Lt})$, which is a scalar by definition, where $f(\cdot)$ is a suitable function known as sorting function, chosen in such a way that neighboring target feature vectors provide almost neighboring values of s_t . Then the mixtures of the GMM are sorted in ascending order of the associated sorting parameter, according to the vector $\mathbf{S} = [s_1, s_2,..., s_M]^T$ with $s_1 \leq s_2 \leq ... \leq s_M$. In this research we simply considered $f(\cdot)$ as the sum of input vector elements and GMM mixtures were sorted in ascending order of the summation of their mean value components.

To compute the likelihood of the input feature vector, in the first step the quantity s_t is scalar quantized by $\mathbf{S}$. Suppose s_i is the result of scalar quantization, with $1 \leq i \leq M$. The index of s_i (i.e., i) is called the central index. In the next step, the input feature vector's likelihood is evaluated using the ordinary method by an extensive local search in the neighborhood of the central index, which includes a subset M_s mixtures out of the entire mixtures $M_s < M$. For example, only the mixtures with indices within the range of $i-k+1$ to $i+k$ may be searched, where k is an offset value ($k = M_s / 2$) .

To achieve better performance for the sorted GMM, always $2k$ mixtures are searched. For example, this means that for the case of $i \leq k$, the first $2k$ mixtures in the GMM are considered for local search, and for $i \geq M-k$ the last $2k$ mixtures are evaluated for the calculation of likelihood. Generally, the computational complexity of this method grows linearly with M_s , which normally is set to be much less than M .

This sorted GMM method can be applied to any GMM, such as a UBM, without any further training process. However, the performance can be further enhanced if the following optimization algorithm is used to optimize the GMM:

Step 1. *Initialization*: Set $n = 0,\ M_n = M_i$, where M_i is the initial GMM. Calculate the sorting parameter related to each mixture and sort the GMM in ascending order of the sorting parameter.

Step 2. *Likelihood Estimation*: Calculate the likelihood of the entire training database with M_n mixtures using the sorted GMM method.

Step 3. *GMM Adaptation*: Compute the M_{n+1} GMM. This is done simply by adapting each mixture from M_n using associated training vectors found in the previous Step.

Step 4. *Sorting*: Recalculate the sorting parameters related to the mixtures of the new GMM, M_{n+1} , and sort the GMM in ascending order of the sorting parameter. Also, set $n = n+1$.

Step 5. *Termination*: If the total likelihood is higher than a certain threshold (or any other reasonable condition), then stop the algorithm; otherwise go to Step 2.

The memory storage required for the sorted GMM is $(2d+2)/(2d+1)$ times of that needed for the ordinary GMM where d is the feature vector length (the negligible extra storage is required to store the sorting parameter quantization table). Therefore, the proposed method has less memory requirement than the tree structured GMM.

After the optimization stage with the optimized UBM in hand, the speakers' GMMs are simply adapted from the optimized UBM like what is done in the ordinary GMM-UBM training.

4. PERFORMANCE EVALUATION EXPERIMENTS

To evaluate the performance of the proposed method several experiments were performed and the results are compared with the competitive schemes such as SBM-SGMM method. The following subsections present more information in this regard.

4.1 Database

The speaker verification experiments were conducted using a set of TV recorded speech database that recorded by the authors [8]. The database is a collection of conversational speech in Farsi, recorded from different channels of Iranian Broadcasting TV using a Winfast® TV card installed on a PC. Recordings were done when the speakers talked in noise free studios and there were no crosstalks or any musical background. The speech signals were recorded with PCM 11025 Hz, 16 bit and mono format. Ninety minutes of speech from 100 male speakers used for the UBM training. About three minutes speech samples for a set of separate 90 male speakers were also recorded to form the target speakers in the test stage. Two minutes of target speakers' speeches reserved for the speakers model adaptation and the last one minutes of speeches were applied in the test procedure.

4.2 Evaluation Measure

The evaluation of the speaker verification system is based on detection error tradeoffs (DET) curves, which show the tradeoffs between false alarm (FA) and false rejection (FR) errors. We also used detection cost function (DCF) defined as [9]

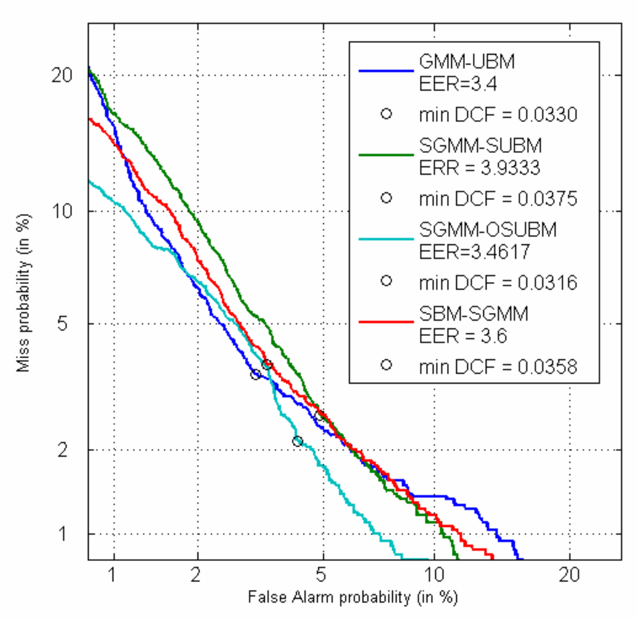

Figure. 1 - Comparison of DET curves of four different GMM based speaker verification systems which use 15 and 3 seconds of speech segments for the training and test, respectively.

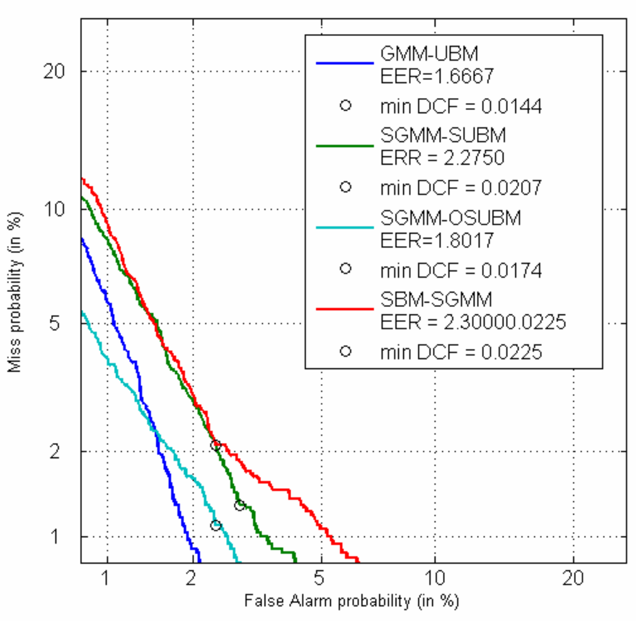

Figure 2 - Comparison of DET curves of four different GMM based speaker verification systems which use 45 and 7 seconds of speech segments for the training and test, respectively.

$$DCF = C_{miss}.E_{miss}.P_{t\arg et} + C_{fa}.E_{fa}(1 - P_{t\arg et})\ . \quad (1)$$

where P_{target} is the a priori probability of target tests with $P_{target} = 0.01$ and the specific cost factors $C_{miss} = 10$ and $C_{fa} = 1$. So the point of interest is shifted towards low FA rates.

4.3 Experimental Setup

In the experiment stage four systems were trained and compared including GMM-UBM, SBM-SGMM, sorted GMM-UBM (a sorted GMM adopted from GMM-UBM which hereafter will be known as SGMM-SUBM) and the optimized sorted GMM-UBM using the aforementioned algorithm which hereafter will be entitled SGMM-OSUBM.

It is noteworthy that the background models in all systems were trained using the 100 speakers subset of the database. Due to the computation and memory restrictions of the computer used for the experiments, the UBM is trained in several stages by producing multiple UBMs from partial sections of this subset at the first stage and then producing data from these models to train the final UBM in the second stage. The data used for the training of the final UBM was also used for the training of the SBM model.

In all tests variants GMMs of order 64 were employed during the experiments reported in this paper according to the findings reported in [10]. Moreover, the applied SBM-SGMM systems have 1-16-64 nodes tree structure.

The offset value, k, was set to 8 for the SGMM-SUBM and SGMM-OSUBM systems, $M_s = 16$. Furthermore, due to inherent sequential nature of the sorting GMM and the optimization algorithm, this algorithm was implemented directly using the optimization of the trained UBM with the entire training feature vectors extracted from the 100 speakers subset of the database. Speech samples for the 60 speakers subset were used for the test stage which was conducted according to the NIST guidelines [9]. No speaker overlap exists between these subsets. The sorting function for the sorting GMM was selected simply as the summation of the feature vectors' elements and mixture models were sorted in ascending order of the summation of the mean values of each mixture. The ratios between target and impostor trials are chosen to be 1:10 and 33000 verification trials from those 60 speakers are used in the test stage.

The SBM-SGMM in these experiments benefited from a nonuniform tree with 1-21-21-21 leaf nodes (this configuration demonstrated the best performance for the current database [8]), and its execution time for the test stage was about 2.7 times faster than that of the GMM-UBM method, while the SGMM-SUBM and the SGMM-OSUBM methods runs about 4 times faster than the GMM-UBM method.

Figs. 1 and 2 show the results of experiments for the tests which use 15 and 45 seconds of training speech utterances of the 60 speakers. In these experiments the duration of test utterances was 3 (7) seconds, respectively. It can be seen that the SGMM-OSUBM method provides superior performance compared to the SBM-SGMM and its superiority is in longer training and test speeches. The rather inferior performance of the SGMM-SUBM compared to the SGMM-OSUBM method was predictable since the sorting criterion and structure was not considered during the training of UBM and associated speakers' GMMs which are used by this method. As the figures show the proposed optimization algorithm has a productive effects on the sorting GMM and enhanced its performance.

Furthermore, these figures confirm that the SGMM-OSUBM and the GMM-UBM performances are almost comparable. This may seem to be strange in the first glance because the latter method is benefited from a brute-force full search while the former just uses a partial search. The reason for this is two-folds. Firstly, to alleviate the personal computer memory and processing power restrictions and to be able to train a UBM for 90 minutes of speech, in our experiments the UBM was trained in three stages by training several UBMs using different subsets of 100 male speakers' utterances and then the models are used to produce data for the next stage of UBM training. This makes the final UBM somehow non-optimum which degrades its performance. The second reason can be explained as the potential of sorting GMM for soft clustering of mixtures with close phonetics statistics as neighboring mixtures. Since in this method only part of the GMM is searched for scoring it may end up with better scoring in comparison with the GMM-UBM method in some occasions.

From computational complexity point of view the computational cost of GMM-UBM is in the order of $O(M+C)$ and for SBM-SGMM used in this study it is in the order of $O(M_1+M_2+C)$, where M_1 and M_2 are the average fan out of for each node in the second and leaf level of the tree. This cost for sorted GMM-UBM is only in the order of $O(M_s+C)$. So for the current example, where $M=64, C=5, M_1=4, M_2=16,\ M_s=16$ (i.e., $k=8$), the computational complexity of GMM-UBM, SBM-SGMM, and SGMM-OSUBM are in the order of 69, 25, and 21, respectively. It is noteworthy that SGMM-OSUBM computational cost increases almost linearly for higher values of k.

5. CONCLUSIONS

In this paper a novel GMM structure called sorted GMM is introduced which is benefited from a fast scoring capability while its memory requirement is just marginally higher than ordinary GMM. Also, the training and optimization schemes of GMMs for the proposed method are described. The use of the proposed method in speaker verification framework is outlined. The performance of a speaker verification system based on the new method was compared with other GMM based speaker verification systems experimentally. The results of experiments prove that the optimized sorted GMM presents a desirable performance and it outperforms the already known SBM-SGMM which itself provides an acceptable performance for fast scoring GMMs while its computational cost is less than that of the SBM-SGMM. Application of sorted GMM in combination with other processing scheme is the subject of future research.

ACKNOWLEDGEMENTS

The authors would like to acknowledge supports of Iranian Research Institute for Electrical Engineering (IRIEE) in the course of this research.

REFERENCES

[1] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Trans. on Speech Audio Processing*, vol. 3, no. 1, pp. 72-83, Jan. 1995.

[2] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19-41, Jan. 2000.

[3] J. Mclaughlin, D. A. Reynolds, and T. Gleason, "A study computation speed-ups of the GMM-UBM speaker recognition system," in *Proc. Eurospeech'99*, pp. 1215-1218, 1999.

[4] K. Shinoda and C. H. Lee, "A structural Bayes approach to speaker adaptation," *IEEE Trans. Speech Audio Processing*, vol. 9, no. 3, pp. 276 287, May 2001.

[5] B. Xiang and T. Berger, "Efficient text-independent speaker verification with structural Gaussian mixture models and neural network," *IEEE Trans. Speech Audio Processing*, vol. 11, no. 5, pp. 447-456, Sept. 2003.

[6] R. Saeidi, H. R. Sadegh Mohammadi, M. Khalaj Amirhosseini, "An efficient GMM classification post-processing method for structural Gaussian mixture model based speaker verification," *to be Presented at ICASSP'06*, Toulouse, France, May 2006.

[7] H. R. Sadegh Mohammadi and W. H. Holmes, "Low cost vector quantization methods for spectral coding in low rate speech coder," in *Proc. of The 1995 International Conf. on Acoustic, Speech, and Signal Processing, ICASSP'95*, vol. 1, pp. 720-723, Detroit, Michigan, US, 1995.

[8] R. Saeidi, H. R. Sadegh Mohammadi, and M. Khalaj Amirhosseini, "Efficient GMM-UBM system in text independent speaker verification using structural Gaussian mixture models," in *Proc. International Symp. of Telecommunications, IST2005,* vol. 1, pp. 39-44, Shiraz, Iran, Sept. 2005.

[9] *The NIST Year 2000 Speaker Recognition Evaluation*, http://www.nist.gov/speech/tests/

[10] R. Saeidi, H.R. Sadegh Mohammadi, and M. Khalaj Amirhosseini "Study of model parameters effects in adapted Gaussian mixture models based text independent speaker verification", in *Proc. International Symp. of Telecommunications, IST2005*, vol. 1, pp. 387-392, Shiraz, Iran, Sept. 2005.