STUDIA UNIV. "BABEŞ–BOLYAI", MATHEMATICA, Volume **XLIX**, Number 1, March 2004

UNSUPERVISED CLASSIFICATION FOR DESIGNING SPEAKER IDENTIFICATION SYSTEMS

MARGIT ANTAL AND ANNA SOÓS

Abstract. We compare recognition performance of Vector Quantization method (VQ) and Gaussian Mixture Modeling method (GMM) in normal speech conditions. We performed measurements to emphasize the relationship between the size of models –number of clusters or number of components of mixtures– and size of the system. We also conclude that the VQ method is a particular case of the GMM method. The results show that the VQ sometimes overperfoms GMM which has a serious shortcoming, particularly when a mixture distribution consists of several overlapping distributions.

1. **Introduction**

Unsupervised classification is also known as data clustering, which is a generic label for a variety of procedures designed to find natural groupings, or clusters, in multidimensional data based on measured or perceived similarities among the number of clusters, the clusters' shapes and the clusters' sizes. In this article we applied unsupervised classification for designing speaker identification systems and we performed several measurements to show the relationship between number of clusters, used to represent speakers' models and identification system's accuracy.

The goal of a speaker identification system is to automatically determine a speaker's identity using an utterance from the speaker. Such a system may be text-dependent—when the speaker must pronounce a text chosen randomly by the system from a fixed vocabulary—, or may be text-independent, when an arbitrary text is allowed to be uttered. Our system was developed for the text-independent case.

Several methods were studied for text-independent speaker identification systems including Vector Quantization methods (VQ) [1], [2], [3], Gaussian Mixture Model method (GMM) [4] and Hidden Markov Models [6]. These methods belong to the model-based approach. For each speaker a statistical model is created to characterize the speaker's voice. These statistical models do not contain any information about interspeaker variabilities.

In this article we try to show that the Vector Quantization method and the Gaussian Mixture method are both based on unsupervised classification, and the VQ method can be viewed as a particular case of the mixture decomposition method.

Received by the editors: 01.10.2003.

Another observation we make is that for speech data, we use clustering only for reduction of the amount of data. Our objective is to find a reduced set of prototypes that best approximate the original set of features and not to find separable clusters (perhaps no any such cluster exists in speech). So we can conclude that there is no significant difference between K-means clustering algorithms developed by the pattern recognition community and the LBG clustering algorithm described in the speech processing and other communications literature [13].

Clustering can be used not only for separating the data into clusters but also for organising a large amount of data. There are hundreds of clustering algorithms in the literature which can be divided in two main categories: square-error iterative partitional clustering and agglomerative hierarchical clustering. In this article we used only the first approach, so we will describe only this one. This type of clustering algorithms attempt to obtain partitions which minimize the within-cluster scatter or maximize the between-cluster scattering [8].

The partitional clustering algorithm determines a partition of n So for clusteringpatterns in a D-dimensional metric space into M ($M < n$) clusters, such that the patterns in a cluster are more similar to each other than to patterns in different clusters. It is a hard problem to determine the optimal clusters' number (M) even when the type of data is known. In this article we performed some measurements to show how the identification system accuracy is influenced by clustering type and the number of clusters used.

2. VQ-based Speaker Identification

In the VQ-based speaker identification system each speaker is represented by a codebook created from some training data uttered by the speaker. Each speaker's model is created in two steps:

- Consider some training data (utterance) from the speaker and extract some type of feature vectors (MFCC [13], LPCC [13])
$$\{x_1, x_2, \dots, x_n\} \qquad x_i \in \mathbb{R}^D.$$
- Cluster the feature vectors into a fixed number of clusters $\{C_1, C_2, \dots, C_M\}$, where $M < n$. Take the centroid of each cluster and form a set of M vectors, named also code vectors. This set of code vectors is called codebook and this is the model of a speaker.

This type of speaker identification system is based on square-error clustering. The objective is to obtain a partition that, for a fixed number of clusters minimizes the square-error. The set of n patterns in D dimensions has somehow been partitioned into M clusters $\{C_1, C_2, \dots, C_M\}$ such that cluster C_k has n_k patterns (feature vectors) and each pattern is in exactly one cluster, so that $\sum_{k=1}^{M} n_k = n$. The mean vector, or center of cluster C_k is defined as the centroid of the cluster:

$$m^{(k)} = \frac{1}{n_k} \sum_{i=1}^{n_k} x_i^{(k)}, \tag{1}$$

where $x_i^{(k)}$ is the ith pattern belonging to cluster C_k [8]. The square-error for cluster C_k, also called within-cluster variation is:

$$e_k^2 = \sum_{i=1}^{n_k} \left(x_i^{(k)} - m^{(k)}\right)^T \left(x_i^{(k)} - m^{(k)}\right). \tag{2}$$

The square-error for the clustering is defined as:

$$E_M^2 = \sum_{i=1}^{M} e_k^2 \tag{3}$$

The objective of this clustering is to find a partition that minimizes (3).

The role of vector quantization (clustering) is to reduce the amount of data and to model the distribution of the feature vectors. The problem of automatically separating training data into groups representing classes is solved by a clustering algorithm. A comparison of clustering algorithms in a VQ-based speaker identification system was made by [5] and the results were that the accuracy of identification of a system generally is not influenced by the clustering algorithm, but is influenced by the number of clusters (codebook size) chosen. So for clustering any efficient and fast algorithm can be used.

The identification procedure can be performed in two ways:

1. comparing the sequence of feature vectors extracted from the unknown speaker utterance $\{x_1, x_2, \dots, x_T\}$ with all N models (codebooks) in the speaker database [1].
2. forming a codebook from the sequence of these feature vectors and comparing the resulting codebook with the codebooks from the speaker database [3].

For case 1 the identification procedure can be formulated as follows:

Consider a speaker idenification system with N known speakers. We define the codebook for the i^{th} speaker as

$$\lambda_i = \left(m_i^{(1)},\ m_i^{(2)},\ \dots, m_i^{(M)}\right), \quad i = 1, 2, \dots, N$$

where $m_i^{(k)}$ is defined by (1).

1. Extract the set of features from the unknown speaker utterance.

$$X = \{x_1, x_2, \dots, x_T\}, \qquad x_i \in \mathbb{R}^D$$

2. For every model $\lambda_i, \quad i = \overline{1, N}$ compute the distortion

$$d(X,\ \lambda_i) = \frac{1}{T} \sum_{k=1}^{T} \min_{j=1..M} d_E(x_k, m_i^{(j)}),$$

where d_E is the Euclidean metric defined in R^D.

3. Identify the speaker as the one with the smallest distortion:

$$Id = \arg \min_{i=1\dots N} \{d\left(X,\ \lambda_i\right)\}$$

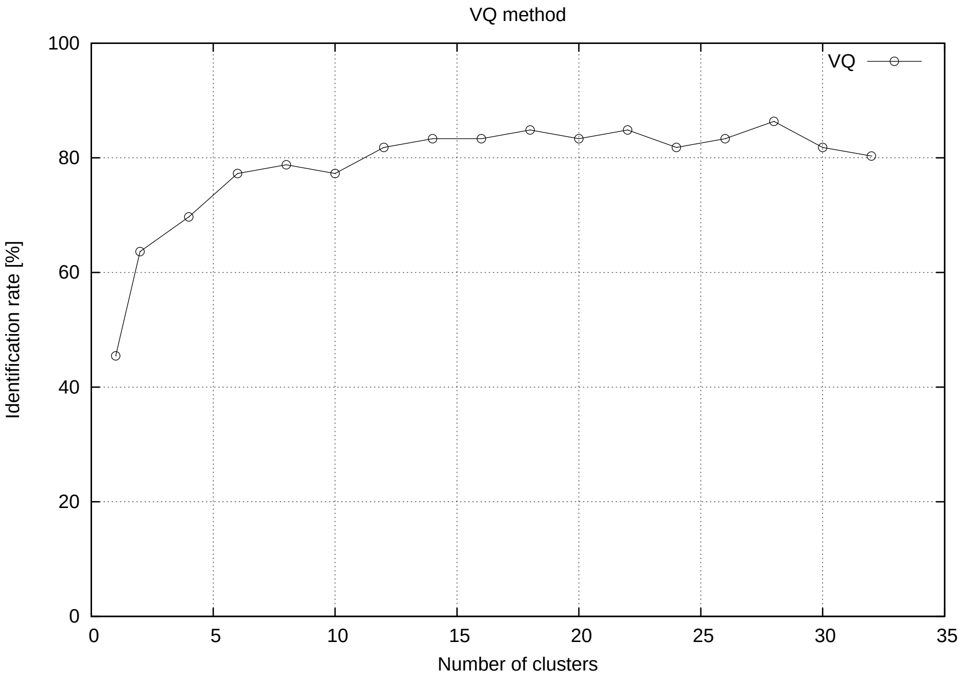


FIGURE 1. Identification rate vs. number of clusters

For case 2 the identification procedure is almost identical to case 1. The only difference is in step (1), where after the feature extraction step is made a codebook from features and this codebook is used in step (2) for calculating the distortions with known speaker models. In this case the algorithm uses a reduced number of distance calculation but performs a clustering to obtain the codebook.

In the application used for measurements we used case 2 for the identification. We trained systems with number of clusters

$$M \in \{1, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32\}$$

and measured the identification rates achieved by these systems. The results are represented in figure 1. The parameters of the identification system are presented in the Section 4.

3. GMM-based Speaker Identification

3.1. **Preliminaries.** Since the primary speaker-dependent information conveyed by the spectrum is about vocal tract shapes, we want to use a speaker model that captures the characteristic vocal tract shapes of a person's voice as manifested in spectral features.

In the statistical speaker model a speaker can be treated as a random source producing the observed feature vectors. The random speaker source is formed by a set of hidden states corresponding to characteristic vocal tract configurations. When the random source is in a particular state, it produces spectral feature vectors from that particular vocal tract configuration. The states are called hidden because we can observe only the spectral feature vectors produced, not the underlying states that produced them. Each state produces spectral feature vectors according to a multidimensional Gaussian probability density function (*pdf*) with a state dependent mean and covariance [4]. The *pdf* for the state i and feature vector x can be expressed as

$$b_i(x) = \frac{1}{(2\pi)^{D/2} |\sum_i|} e^{-\frac{1}{2}(x-\mu_i)^T \sum_i^{-1}(x-\mu_i)} \tag{4}$$

where

- x is a D-dimensional feature vector, $x \in R^D$
- μ_i is the state mean vector, $\mu_i \in R^D$
- Σ_i is the state covariance matrix

The mean vector represents the expected spectral feature vector from the state, and the covariance matrix represents the correlations and variability of spectral features within the state.

The produced feature vector depends on the parameters of the current state $(\mu_i,\ \Sigma_i)$ and the process governing what state the speaker model occupies at any time is modeled as a random process. The following discrete *pdf* associated with the M states describes the probability of being in any state

$$\{p_1, p_2, \ldots, p_M\}, \qquad \text{where } \sum_{i=1}^{M} p_i = 1, \tag{5}$$

and a discrete pdf describes the probability that a transition will occur from one state to any other state,

$$a_{ij} = P(i \to j), \qquad i, j = \overline{1, M} \tag{6}$$

The above definition of the statistical speaker model is known as Hidden Markov Model (HMM) [15]. The HMMs are capable of describing a complex statistical process.

Because our goal is to build speaker's models for text independent speaker recognition we can simplify the statistical speaker model by setting the transition probabilities a_{ij} equal to $1/M$. This means that all state transitions are equally likely.

In following sections we will call each state a component.

3.2. **The Gaussian Mixture Speaker Model.** A Gaussian mixture density of a feature vector x, $\ x \in \mathbb{R}^D$, given the parameter vector λ is a weighted sum of M component densities, and is given by the equality:

$$p(x|\lambda) = \sum_{i=1}^{M} p_i \cdot b_i(x) \tag{7}$$

where

- x is a D-dimensional feature vector
- $b_i(x) \quad i = \overline{1, M}$ are the component densities
- $p_i \quad i = \overline{1, M}$ are the mixture weights.

Each component density is a D-variate Gaussian function defined by the equation (4) with mean vector μ_i and covariance matrix Σ_i and the mixture weights satisfy the constraint

$$\sum_{i=1}^{n} p_i = 1$$

The complete Gaussian mixture density is parameterized by the mean vectors, covariance matrices and mixture weights from all component densities.

These parameters are collectively represented by the symbol:

$$\lambda = (p_i, \mu_i, \Sigma_i), \quad i = \overline{1, M}$$

There are two principal advantages for applying Gaussian mixture densities as a representation of speaker identity. The first is the intuitive notion that the individual component densities of a multi-model density may model some underlying set of acoustic classes. These acoustic classes reflect some general speaker-dependent vocal tract configurations that are useful for characterizing speaker identity. The second advantage of using Gaussian mixture densities for speaker identification is the empirical observation that a linear combination of Gaussian basis functions is capable of representing a large class of sample distributions. One of the powerful attributes of GMM is its ability to form smooth approximations to arbitrarily-shaped densities.

3.3. **Applying the model.** With the GMM as the speaker representation we can then apply this model to speaker identification. The identification system is a maximum likelihood classifier. For a reference group of N speaker models $\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$, the objective is to find the speaker identity $\hat{s}$ whose mode has the maximum posterior probability for the input feature vector sequence

$$X = \{x_1, x_2, \ldots, x_T\}$$

The minimum-error Bayes' rule for this problem is

$$\hat{s} = \arg \max_{1 \leq s \leq N} P(\lambda_s | X) = \arg \max_{1 \leq s \leq N} \frac{p(X|\lambda_s)}{p(X)} \cdot P(\lambda_s) \tag{8}$$

Assuming equal probabilities of speakers, the $P(\lambda_s)$ and $p(X)$ are constant for all speakers and can be ignored. Equation (8) becomes

$$\hat{s} = \arg \max_{1 \leq s \leq N} p(X|\lambda_s)$$

Assuming independence between observations the decision rule for the speaker identity becomes

$$\hat{s} = \arg \max_{1 \leq s \leq N} \prod_{t=1}^{T} p(x_t|\lambda_s) \tag{9}$$

where

- T is the number of feature vectors
- $p(x_t|\lambda_s)$ is given in equation (7)

Because the logarithm is monotonically increasing, (9) becomes

$$\hat{s} = \arg \max_{1 \le s \le N} \sum_{t=1}^{T} \log p(x_t|\lambda_s)$$

3.4. **Estimating GMM parameters.** Given a training speech from a speaker, the goal of speaker model training is to estimate the parameter vector λ for GMM. We will use Maximum Likelihood (ML) estimation technique. The aim of ML estimation is to find the model parameters which maximize the likelihood of the training data.

For a sequence of T training feature vectors

$$X = \{x_1, x_2, \ldots, x_T\}$$

the GMM likelihood can be written as than model with fewert

$$p(X|\lambda) = \prod_{t=1}^{T} p(x_t|\lambda)$$

The ML parameters can be estimated by using a specialized version of the expectation-maximization (EM) algorithm. The basic idea of the EM algorithm is beginning with an initial model λ, to estimate a new model $\overline{\lambda}$, such that $p(X|\overline{\lambda}) \ge p(X|\lambda)$. The new model then becomes the initial model for the next iteration.

On each EM iteration the following estimates are calculated:

Mixture weights::

$$\overline{p_i} = \frac{1}{T} \sum_{t=1}^{T} p(i|x_t, \lambda)$$

Means::

$$\overline{\mu_i} = \frac{\sum_{t=1}^{T} p(i|x_t, \lambda) \cdot x_t}{\sum_{t=1}^{T} p(i|x_t, \lambda)}$$

Covariances::

$$\overline{\Sigma_i} = \frac{\sum_{t=1}^{T} p(i|x_t, \lambda) \cdot x_t x_t^T}{\sum_{t=1}^{T} p(i|x_t, \lambda)} - \overline{\mu}_i \bar{\mu_i}^T$$

If we are using diagonal covariance matrices, we need to update only the diagonal elements in the covariance matrices. For an arbitrary diagonal element σ_i^2 of the covariance matrix of the i^{th} mixture, the variance estimates become:

$$\bar{\sigma_i}^2 = \frac{\sum_{t=1}^{T} p(i|x_t, \lambda) x_t^2}{\sum_{t=1}^{T} p(i|x_t, \lambda)} - \overline{\mu_i}^2$$

The aposteriori probability for component i is given by

$$p(i|x_t, \lambda) = \frac{p_i \cdot b_i(x_t)}{\sum_{k=1}^{M} p_k \cdot b_k(x_t)}$$

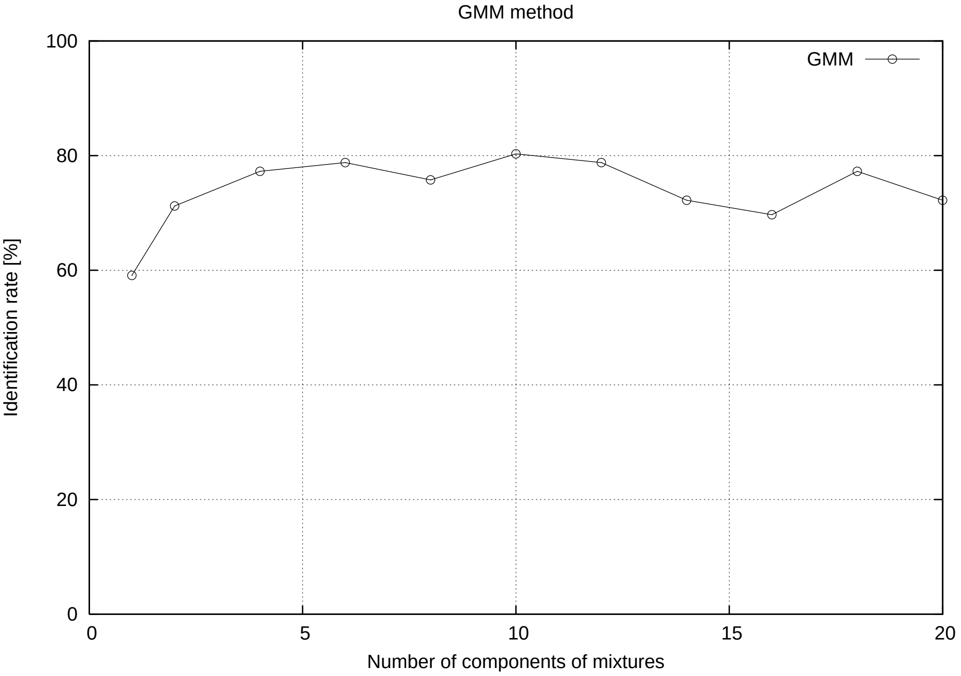


Figure 2. Identification rate vs. mixture numbers

Each of the equations involves $p(i|x_t, \lambda)$,which can be interpreted as “fuzzy membership” of x_t to Gaussian i.

Initialization of the GMM models

First, the order M of the model must be large enough to represent the feature distributions. Second, the type of the covariance matrices for the mixture distributions needs to be selected. Diagonal covariance matrices simplify the implementation and are computationally more feasible than models with full covariances. The EM algorithm guarantees to find a *local maximum* likelihood model regardless of the initialization, but different initialization can lead to different local maxima. Usually the means are initialized with centroids of clusters obtained with k-means algorithm and for covariance matrices can be used as initial value the identity matrices.

3.5. **Experimental results.** In the first experiment we tested how the mixtures' components number (model order) influence the identification accuracy. The results are given in figure 2.

In the second experiment we tested if there exists a relationship between the identification system size –speakers known by the system– and the mixture components. We measured the identification rate for systems with mixture's components 1, 2 and 4, increasing the system size (number of speakers) from 1 to 66. As the

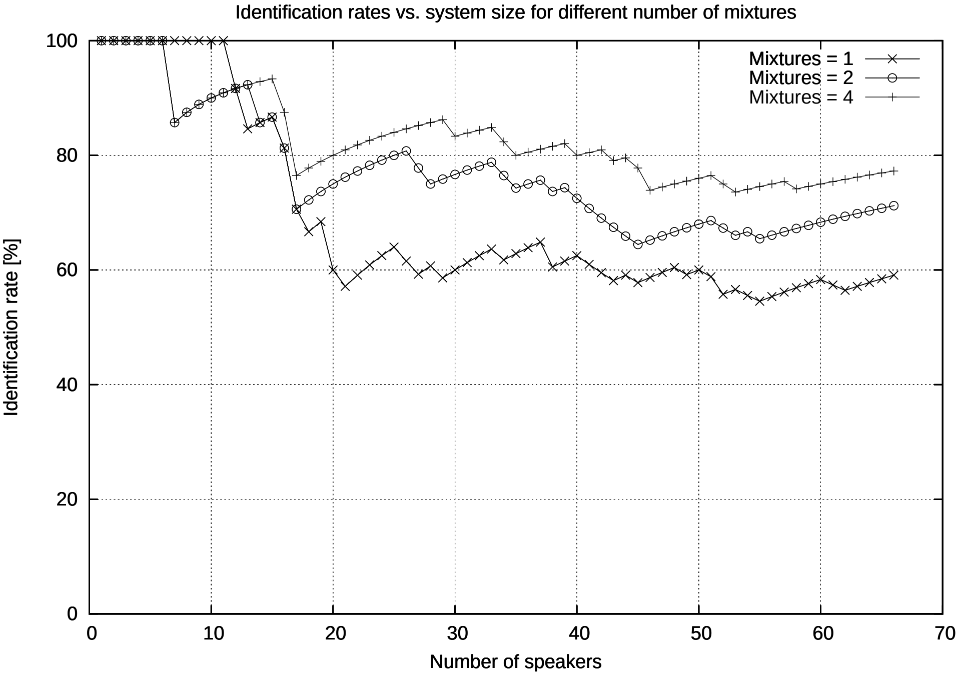


FIGURE 3. Identification rates vs. system size obtained for systems with mixture components 1, 2 and 4

number of speakers increases the model with more mixture components achieves a better performance than the model with fewer components.it was made a The results are shown in figure 3.

4. Conclusions

All experiments were done with speech collected from 66 speakers, 29 Hungarian native speakers and 37 Romanian native speakers. 45 of 66 were female speakers and 11 were male speakers. The ages of speakers vary from 14 to 60. The speech was recorded with at least four types of microphones on anonymous soundcards without laboratory conditions. The sampling rate was 16 kHz with 16 bits/sample. Before feature extraction stage a preprocessing was made with direct component (DC) removal and a high emphasis filtering with

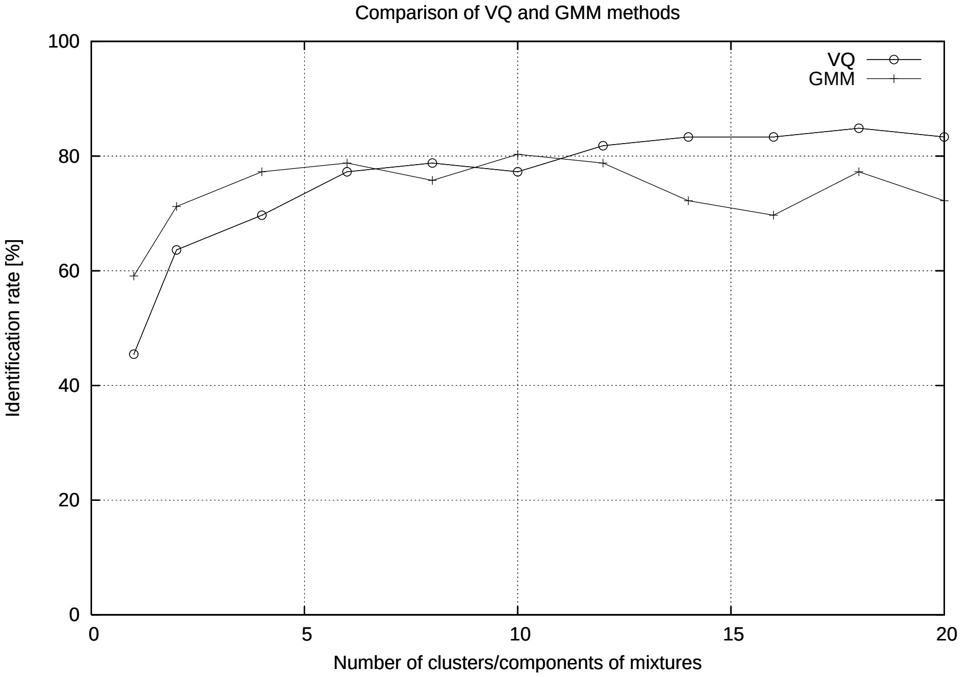
$$H(z) = 1 - 0.95 \cdot z^{-1}$$

and finally, performed a short-term mel-cepstrum analysis with 30 ms Hamming window, with 10 ms shift. The number of mel-cepstral coefficients was 12. For training purposes we used 30s speech and for identification a new set of 1s speech collected from each speaker.

In the stage of estimating the GMM models we initialized the parameters of the model with the following values:

- $p_i = \frac{1}{M}$,
- μ_i were obtained with the clustering algorithm used in the VQ model too
- for Σ_i we used diagonal covariance matrix initialized with the identity matrix, $i = \overline{1, M}$.

Our goal was to compare the VQ method with the GMM method so we used the same features obtained from the same speech database and the same clustering for both methods. Our expectation was that the GMM method will overperform the VQ method, but this is not the case for all values of M. The following figure shows the results obtained for the two systems.



The GMM models' construction is more computer-time consuming and may have a serious shortcoming, particularly when a mixture distribution consists of several overlapping distributions [11].

References

[1] R. K. Soong, A. E. Rosenberg, B. H. Juang, L. R. Rabiner, *A Vector Quantization Approach To Speaker Recognition*, AT&T Technical Journal, 66(1987), 14-26.

[2] J. P. Campbell, *Speaker Recognition: A Tutorial*, Proc. IEEE, vol. 85, no. 9, 1997, 1437-1462.

[3] T. Kinnunen, P. Franti, *Speaker Discriminative Weighting Method for VQ-based Speaker Identification*, Proc. 3^{rd} International Conference on audio- and video-band biometric person authentication, Halmstad, Sweden, 2001, 150-156.

[4] D. A. Reynolds, *Automatic Speaker Recognition Using Gaussian Mixture Speaker Models*, The Lincoln Laboratory Journal, Vol. 8, No. 2, 1995.

[5] T. Kinnunen, Teemu Kilpelainen, Pasi Franti, *Comparison of Clustering Algorithms in Speaker Identification*, Proc. LASTED International Conference, Signal Processing and Communications, Marbella, Spain, 2000, 222-227.

[6] J. M. Naik, L. P. Netsch, G. R. Doddington, *Speaker Verification over Long Distance Telephone Lines*, Proc. ICASSP'89, pp. 524-527, May, 1989.

[7] Bojan Nedic, Herve Bourlard, *Recent Developments in Speaker Verification at IDIAP*, IDIAP-RR 00-26, September 2000.

[8] Anil K. Jain, Robert P. W. Duin, Jiangchang Mao, *Statistical Pattern Recognition: A Review*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No.1, 2000.

[9] L. Wang, K. Chen, H. Chi, *Capture Interspeaker Information With a Neural Network for Speaker Identification*, IEEE Transactions on Neural Networks, Vol. 13, No. 2, 2002.

[10] R. O. Duda, P. E. Hart, D. G. Stork, *Pattern Classification*, John Wiley&Sons, 2001.

[11] Keinosuke Fukunaga, *Introduction to Statistical Pattern Recognition*, Second Edition, Morgan Kaufmann, 1990.

[12] F. Jelinek, *Statistical Methods for Speech Recognition*, MIT Press, Third Edition, 2001.

[13] J. R. Deller, Jr. J. H. L. Hansen, J. G. Proakis, *Discrete-Time Processing of Speech Signals*, John Wiley&Sons, 2000.

[14] A. K. Jain, R. C. Dubes, *Algorithms for Clustering Data*, Englewood Cliffs, Prentice Hall, 1988.

[15] L. R. Rabiner, B. H. Juang, *Fundamentals of Speech Recognition*, Prentice-Hall, Englewood Cliffs, 1993.

[16] L. R. Rabiner, *A tutorial on hidden Markov models and selected applications in speech recognition*, Proceedings of IEEE, 77(2), 1989, 257-286.

Sapientia Hungarian University of Transylvania
E-mail address: manyi@ms.sapientia.ro

Babeş-Bolyai University Cluj-Napoca
E-mail address: asoos@math.ubbcluj.ro