Efficient Text-Independent Speaker Identification using Optimized Hierarchical Mixture Clustering

B. Bharathi
Department of CSE
SSN College of Engineering
Kalavakkam

P. Jansi Rani
Department of CSE
SSN College of Engineering
Kalavakkam

ABSTRACT
Conventional Speaker Identification(SI) Systems uses individual Gaussian Mixture Models(GMM) for every speaker. If this method used for the large population Speaker identification systems, then during identification, likelihood computations between an unknown speaker's test feature vectors and speaker models has become a time-consuming process. This approach also increases the computational complexity and the training time. To overcome these problems this paper proposes an clustering method named as, Optimized Hierarchical mixture Clustering (OHMC) in combination with Maximum A posteriori(MAP) and Maximum Linear Log Regression (MLLR) Adaptation. In this approach, during training phase, speaker independent model (Universal Background Model) was generated and then individual speaker models has been built using supervised adaptation. Then OHMC method used to group the similar speakers. During testing phase, log-likelihood needs to be calculated only for the cluster which is scored high among all the clusters, and subsequently for all speakers within that cluster. This method uses Kullback - Leibler divergence as the distance measure. Hence this combined approach performs well when compared to other conventional approaches by reduced the computational complexity and increased the identification speed.

General Terms
Pattern recognition, speech processing, speaker recognition.

Keywords
Speaker Identification, Optimized Hierarchical Mixture Clustering, MAP and MLLR Adaptation.

1. INTRODUCTION
The voices of different individuals do not sound alike. The ability of recognizing the person solely from his voice is known as speaker recognition. Speaker recognition refers to two fields.[7] 1. Speaker Identification (SI) - Determine which voice sample from a set of known voice samples best matches the characteristics of an unknown input voice sample. 2. Speaker Verification (SV) - Determine whether an unknown voice is from a particular enrolled speaker or not. Of the various speaker modeling techniques, the Gaussian Mixture Model - Universal Background Model (GMM-UBM) based approach and MAP adaptation of the speaker models has shown to be very successful in accurately identifying speakers from a large population and is presently state-of-the-art technique. [1] In [2], the author quoted as the Gaussian mixture model-universal background model approach is the dominant one in text-independent speaker recognition. The idea of using UBM is in order to capture the general characteristics of a population and then adapting it to the individual speaker. A standard approach in estimating the parameters of the GMM-UBM (weights, mean vectors, and covariance matrices) is to use the Expectation Maximization (EM) algorithm.[5] Generally mean vectors of the GMM-UBM are adapted because only mean adaptations gives the best result. Since, each speaker model is adapted from the GMM-UBM by tuning only its mean parameter (without changing weights and covariance matrices); a point to point link exists between each Gaussian component of adapted speaker model and the GMM-UBM.[9] So, it provides good performance even though the speaker-dependent data is small. Hence in this paper GMM-UBM technique was used during training and testing phase to build speaker independent models and MAP and MLLR (Maximum Likelihood Linear Regression) Adaptation used for building speaker specific models. In [9], they have proposed a Hierarchical Mixture clustering algorithm to increase the identification speed and improving the accuracy for large population. However, cluster was not proper for all mixtures, because this clustering process was fully based on number of speakers in the the data set and also it takes more time for training. Hence we have proposed Optimized Hierarchical Mixture Clustering(OHMC) algorithm to improve the speaker identification process in all the aspects.

2. SPEAKER IDENTIFICATION APPROACHES

2.1 Gaussian Mixture Model
For text-independent speaker recognition, where there is no need for prior knowledge of what the speaker will say, the most successful likelihood function was GMMs. In text dependent applications, where there is a requirement of strong prior knowledge of the spoken text, additional temporal knowledge can be incorporated by using hidden Markov models (HMMs) for the likelihood functions. To date, however, the use of more complicated likelihood functions, such as those based on HMMs, have shown no advantage over GMMs for text independent speaker detection tasks. Such a system is normally used to measure the effectiveness of novel algorithms in modeling approaches. In this method speakers are modeled with GMMs based on the maximum-likelihood (ML) criterion, which has been shown to outperform several other existing techniques.

For a D-dimensional feature vector, the mixture density used for the likelihood function is defined as follows in Eq (1), [7]

$$P(\frac{x}{\lambda}) = \sum_{i=1}^{M} P_i b_i(\vec{x}) \quad (1)$$

Each component density is a Gaussian function of the form as Eq (2),

$$b_i(\vec{x}) = \frac{1}{2\pi^{\frac{D}{2}}\sigma_i^{\frac{1}{2}}} \exp\{-\frac{1}{2}(\vec{x}-\vec{\mu_i})'\sigma_i^{-1}(\vec{x}-\vec{\mu_i})\}, \quad (2)$$

So, the complete Gaussian mixture parametrized as in Eq (3),

$$\lambda = \{p_i, \mu_i, \sigma_i\} \quad (3)$$

Given a collection of training vectors, maximum likelihood is estimated by using the iterative expectation-maximization (EM) algorithm. The EM algorithm iteratively refines the GMM parameters to monotonically increase the likelihood of the estimated model for the observed feature vectors. Generally, five to ten iterations are sufficient for parameter convergence. The advantages of using the GMM as the likelihood function are, it is computationally inexpensive and is based on a well-understood statistical model. For text-independent tasks it is insensitive to the temporal aspects of the speech and only the underlying distribution of acoustic observations from a speaker has been modeled. The latter is also a disadvantage, because higher-levels of information about the speaker, conveyed in the temporal speech signal were not used by this approach. [3]

2.1 Adapted GMM system

Unlike the standard approach of maximum likelihood training of a model for each speaker, the basic idea in the adaptation approach is to derive the speaker's model by updating the well-trained parameters in the background model via adaptation. This provides a tighter coupling between the speaker's model and background model that not only produces better performance than decoupled models, but also allows for a fast-scoring technique. [3] Like the EM algorithm, the adaptation is a two-step estimation process. The first step is identical to the ``expectation" step of the EM algorithm, where estimates of the sufficient statistics of the speaker's training data are computed for each mixture in the UBM. Unlike the second step of the EM algorithm, in this adaptation method, the ``new" sufficient statistic estimates are then combined with the ``old" sufficient statistics from the background model mixture parameters using a data-dependent mixing coefficient. The data-dependent mixing coefficient is designed so that mixtures with high counts of data from the speaker rely more on the new sufficient statistics for final parameter estimation, and mixtures with low counts of data from the speaker rely more on the old sufficient statistics for final parameter estimation. One possible explanation for the improved performance is that the use of adapted models in the likelihood ratio is not affected by ``unseen" acoustic events in recognition speech.

The adapted GMM approach also leads to a fast-scoring technique. This fast-scoring approach is based on two observed effects. The first is that when a large GMM is evaluated for a feature vector, only a few of the mixtures contribute significantly to the likelihood value. This is because the GMM represents a distribution over a large space but a single vector will be near only a few components of the GMM. Thus likelihood values can be approximated very well using only the top 1-5 best scoring mixture components. The second observed effect is that the components of the adapted GMM retain a correspondence with the mixtures of the background model so that vectors close to a particular mixture in the background model will also be close to the corresponding mixture in the speaker model. [3]

3. SYSTEM ARCHITECTURE

Speaker recognition system has two phases.

1. **Training Phase:** During enrollment, the speaker's voice is recorded and typically a number of features are extracted to form a voice print, template, or model. It is illustrated in Fig(1).
2. **Testing Phase:** In the identification phase, a speech sample or ``utterance" is compared against voice print which is created in training phase. It is illustrated in Fig(2).

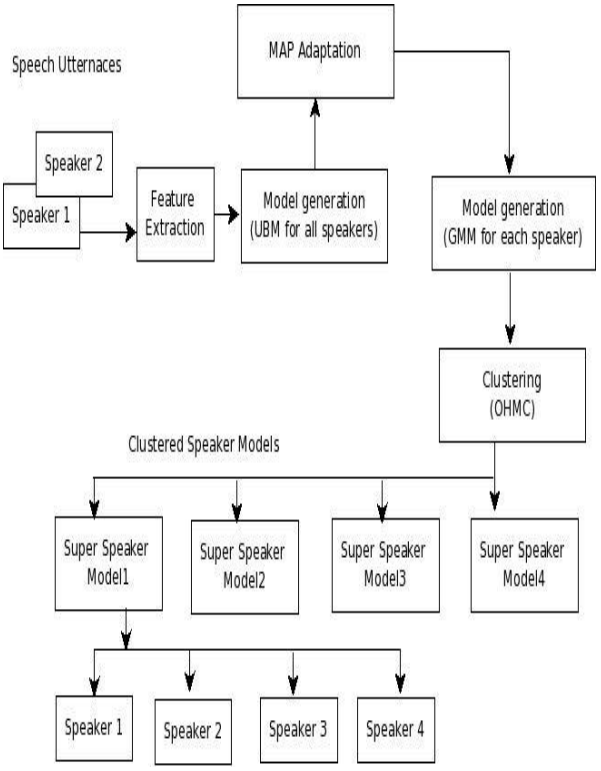

Fig. 1. Training phase of speaker identification

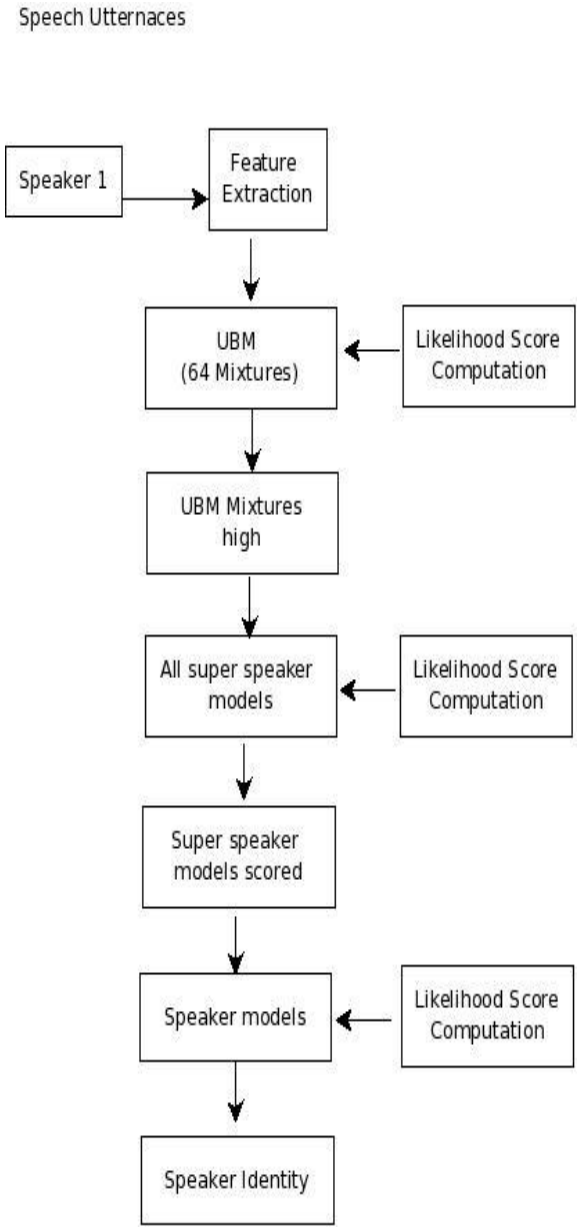

Fig. 2. Testing phase of speaker identification

In training phase, individual speaker models for all speakers was constructed at the beginning. Steps for constructing the models was described below: First, speech samples are selected from corpus. Then MFCC (Mel Frequency Cepstral Coefficients) is extracted from speech sample, which reflects vocal tract shape. It is robust to noise when compared to other features.[6] In [9], they explained the method for extracting the MFCC. They proved that MFCC can be used efficiently for speaker identification with higher identification rates. It compress speech data by eliminating information not pertinent to phonetic analysis E.g., represent an 20 ms of speech audio with 39 real numbers. MFCC tries to mimic the way our ears work by analyzing the speech waves linearly at low frequencies and logarithmically at high frequencies using filter bank processing. Then, the UBM model was constructed using all the speaker's speech utterances. From that model, individual speaker model has been built by MAP+MLLR Adaptation. Then OHMC algorithm was used to form the clusters for every mixture.

In the test phase, speaker identification can be done as follows: First, test speech sample was selected, then feature vectors are extracted from that speech utterance. Each test vector from the speech utterance is set to scored against all UBM Gaussian components, and a small number (typically 5) of the best scoring components in the corresponding speaker-dependent GMMs are taken into consideration. Next, likelihood computation was calculated for clusters(for identifying the best cluster from that mixture) and then for speaker models (for identifying the best speaker model from that cluster). Finally, OHMC approach discloses speaker identity.

4. CLUSTERING

4.1 Motivation for clustering

In the Baseline system (GMM Based), during testing phase, feature vectors are extracted from the test speech utterance. Then, these feature vectors is compared with all the speaker models (For ex.80). Finally, the model which is scored high was considered as the target speaker model. Speaker identification accuracy will be more in this conventional approach, but it comes with the cost of increased computational complexity and also reduced identification speed for large population.

In clustering approach, similar speakers are grouped together based on their parameters. In the test phase, log-likelihood needs to be calculated only for the cluster which is scored high among all the clusters. Hence to speed-up the identification process and to reduce the computational complexity for large population, clustering approach has been proposed.

4.2 Clustering approaches

In conventional system, K-Means clustering method can be used to group the similar speakers. However the problem is, the clustering process fully based on the number of clusters which is initialized at first. So, to avoid this prior initialization, hierarchical clustering approach can also be used.[4] In the proposed method an associated tree was constructed for each mixture of UBM with the corresponding mixture of speakers as the leaf nodes. For the sake of simplicity, the order of UBM and the number of speakers in the system are assumed to be equal to 64 and 80, respectively. Suppose the tree has a middle layer with 5 nodes, we consider the mixtures of these nodes as the mixtures of 5 virtual speakers which hereafter will be called as super-speakers. Let the tree be considered as a uniform tree, hence each node in the middle layer (super-speaker mixture) has 16 associated mixtures of the speakers as leaf node. With such structure, in the test phase after finding the top scoring mixture in the UBM, e.g., the m -th mixture, the associated tree is search down to get the speaker whose mixture provides the best score. Normally, the baseline GMM method needs 80 Gaussian computations for speaker's GMM scoring, the proposed OHMC method needs only 5+16=21 Gaussian computations (for search in middle and leaf layers). [9]

To calculate distance between speakers, we used symmetric KL-divergence which is defined in Eq (4)

$$D(KL)(i,j) = \frac{1}{2}\sum_{k=1}^{K}\left[\frac{\sigma_i^2(k)+(\mu_i(k)-\mu_j(k))^2}{\sigma_j^2(k)}+\frac{\sigma_j^2(k)+(\mu_i(k)-\mu_j(k))^2}{\sigma_i^2(k)}-2\right] \quad (4)$$

4.2 Hierarchical Mixture Clustering Algorithm

The {Termination Condition} is an important property in a hierarchical Clustering algorithm. In [9], the author proposed an Hierarchical Mixture Clustering algorithm with a termination condition based on number of speakers in a data set. However, the algorithm provides better accuracy and speedup in identification speed, the clusters was not proper for all mixtures and it took more time for forming a cluster for each mixture. Hence that algorithm was modified in the termination condition to produce the proper clusters, which is described below:

Initial Steps for every mixture

1. Set the maximum number of speakers clustered in a cluster, maximum leafs of a node in the middle layer, to MaxSpeaker (equal to 16 in this study).
2. Maintain all speaker's mean and variances of single mixture, all_mean_model .
3. Form a distance matrix between all possible pair mixtures of speakers and find the two closest ones (i.e) the minimum value, min_val. The column indexes of these elements will be referred to as i and j speakers.
4. Form the cluster.
    a. Then, keep track of i and j values (i.e) the speaker indexes which are clustered, in a file called rowcol.
    b. Construct the model for this speaker values.
    c. Append this new model's mean and variance to all_mean_model.
    d. Set that row and columns (i, j) mean and variance values are 0.
    e. Calculate the distance again and from this updated distance matrix find the minimum value, min_val2.

For Every Mixture

5. Repeat all steps below until min_val2! =null.
6. First, find the two closest ones (i.e) the minimum value, min_val. The column indexes of these elements will be referred as i and j speakers.
7. If the sum of the length of the members in ith and jth places in a file, row_col is less than MaxSpeaker then, repeat the step 3.
8. If the sum of the length of the members in ith and jth places in a file, row_col is equal to MaxSpeaker, then repeat the step 3 and then include the speaker indexes which are clustered into the final clusters set, final_cluster. Update the row and column values as 0.
9. If the sum of the length of the members in ith and jth places in a file is greater than MaxSpeaker, then select among mixtures i and j the one who has more members as one of the final super speakers and remove the statistic from rowcol and all_mean_model and add the speaker numbes to final_cluster. Update the row (or) column values as 0.

4.3 Optimized Hierarchical Mixture Clustering Algorithm

This OHMC algorithm provides better performance than other approaches in aspects of identification speed, computation complexity and accuracy, which is described below:

For Every Mixture

1. Set the maximum number of speakers clustered in a cluster, i.e., maximum leafs of a node in the middle layer, to MaxSpeaker (equal to 16 in this study).
2. Form a distance matrix between all possible pair mixtures of speakers and find the two closest ones (i.e) the minimum value, min_val. The column indexes of these elements will be referred to i and j speakers.
3. Then, keep track of i and j values(i.e) the speaker indexes which are going to be clustered, in a set called rowco11.
4. Repeat all the steps below.
5. If the sum of the length of the members in ith and jth places in a set, row_col1 is less than MaxSpeaker then, do the step 6 else go to step 7
6. Form the cluster.
    a. Update distance matrix : For updating the values find the minimum values by comparing row and column values. Append this minimum values as new row and new column in distance matrix.
    b. Set the old row and columns (i, j) values in the distance matrix as 0.
7. If the sum of the length of the members in ith and jth places in the set, row_co1l is equal to MaxSpeaker, then repeat the step 6 and then include the speaker indexes which are clustered into the final clusters set, final_cluster else go to step 8.
8. If the sum of the length of the members in ith and jth places in the set is greater than MaxSpeaker, then select among mixtures i and j the one who has more members as one of the final super speakers and remove the statistic from rowcol1 add them to final_cluster.
9. Find the minimum value, min_val2 from the distance matrix. Repeat the above described steps from step 5, until min_val2! =null.

In this way a non-uniform tree structure per mixture of UBM was constructed. It will cluster the speakers differently for each UBM mixture, i.e. a mixture of a particular speaker's GMM may belong to the first super speaker as a leaf node while another mixture of the same speaker belongs to the fourth super speaker as a leaf node, and so on. The objective in this method was to construct a tree per mixture that has 5

nodes in the middle layer (5 super-speaker models). It was done by setting Max_speaker equal to 16. Then, 6 nodes was used instead of 5, because it was observed in the experiments that, 4 out of 64 cases the number mixtures associated with one super-speaker exceeded Max_speaker [9]. In the identification phase, UBM mixtures are evaluated to find the top scoring mixture and this index is used in the associated tree to evaluate the best super-speaker. Then an exhaustive search was performed for the speakers whose mixtures were in the subset of the selected super speaker. As the trained tree structure is non-uniform, the number of Gaussian evaluated for each feature vector may differ from the others.

5. EXPERIMENTAL SETUP AND ANALYSIS

This section is presented the experimental method and the results of the speaker identification system.

5.1 Speech Database

The TIMIT corpus of read speech has been designed to provide speech data for the acquisition of acoustic-phonetic knowledge and for the development and evaluation of automatic speech recognition systems. TIMIT has resulted from the joint efforts of several sites under sponsorship from the Defense Advanced Research Projects Agency - Information Science and Technology Office (DARPA-ISTO). Text corpus design was a joint effort among the Massachusetts Institute of Technology (MIT), Stanford Research Institute (SRI), and Texas Instruments (TI). The speech was recorded at TI, transcribed at MIT, and has been maintained, verified, and prepared for CD-ROM production by the National Institute of Standards and Technology (NIST).[1] It contains a total of 6300 sentences,10 sentences spoken by each of 630 speakers from 8 major dialect regions of the United States. In this database, speech segment length will be approximately 3 seconds.

5.2 Training

We were tested the proposed method in 3 ways: First with 80 female speakers, next with 80 male speakers, then 160 speakers of male and female combination. From 10 speech samples of every 80 speakers , eight speech utterances used for training and two taken for testing. Hence for these speech utterances, Feature Vectors (MFCC) was extracted and speaker specific model has been constructed by the combination of MAP+MLLR Adaptation. Next, OHMC needs to be applied for every mixture (1, 2...64) of all speakers (1, 2...80). It will creates, 5 or 6 clusters for every mixture.

5.3 Testing

In this phase, first the top scoring mixture, from 64 mixtures needs to be found for the test utterance. Then, for that mixture's associated clusters were tested for finding the best cluster out of 5 or 6 clusters. Then speaker models within that top scoring cluster were tested, to find the speaker which is having maximum likelihood.

5.4 Experimental Analysis

Optimized Hierarchical mixture clustering in combination with adaptation method is tested in 4 aspects individually, they are: the training time, identification time, accuracy and computational complexity. When considering the time and computational complexity for training and testing the models, the proposed method results better performance than conventional approaches by using adaptation. According to the speaker identification accuracy, in OHMC, it identified the speakers correctly even when only one top cluster was used. It is also expected that the accuracy will get improved if number of clusters (used for testing) was increased.

5.5 Performance Analysis

The following factors will help to investigate the speaker identification performance between different approaches.

1. Training Time
2. Identification Time
3. Accuracy
4. Computational Complexity

5.5.1 Training Time

Table 1 compares model generation time taken by Conventional GMM approach and Adaptation method.

Table 1. Training time using Adaptation and Conventional GMM

Model generation method	Number of speakers	Training time (Seconds)
Conventional GMM	1	6.81
Adaptation method	1	0.64

Table 1 shows that adaptation method took very less time for model generation than any other conventional approach.

5.5.2 Identification Time

Speaker identification time taken by Conventional GMM approach and different clustering methods have been shown Table 2.

Table 2 Speaker identification time using conventional GMM and two clustering methods

Model generation method	Number of speakers	Speaker identification time (Seconds)
Conventional GMM	80	0.6
	160	1.2
HMC	80	1.2
	160	1.3
OHMC	80	1
	160	1.2

From Table 2, we have identified that when number of speakers increases the identification time also increases. In the case of conventional approaches, identification time was increased twice when the population increases, also took more time than clustering approaches.

5.5.3 Identification accuracy

Table 3, 4, 5 compares speaker identification accuracy taken by different clustering methods.

Female speakers

Table 3 Speaker identification accuracy for female speakers

Clustering type	No. of speakers	No. of speakers identified correctly	% of identification accuracy
HMC	80	67	84
OHMC	80	70	87.5

Male speakers

Table 4 Speaker identification accuracy for male speakers

Clustering type	No. of speakers	No. of speakers identified correctly	% of accuracy
HMC	80	69	86
OHMC	80	70	87.5

Table 5 Speaker identification accuracy for both male and female speakers

Clustering type	No. of speakers	No. of speakers identified correctly	% of accuracy
HMC	160	132	82.5
OHMC	160	137	86

Table 3,4,5 shows that OHMC results out better accuracy than HMC

5.5.3 Computational Complexity - Cluster Generation Time

The cluster generation time for single mixture have been and analysed and tabulated in Table 1.

Table 6 Cluster generation time for single cluster

Cluster Type	No. of Speakers	Cluster generation time for single cluster
HMC	80	3.5 Hrs
	160	9.5 Hrs
OHMC	80	2 seconds
	160	4 seconds

Table 6 shows the significant advantage of OHMC over HMC. Because it reduces the cluster generation time from hours to seconds. From performance analysis results we inferred that, Optimized Hierarchical Mixture Clustering outperforms than conventional GMM and HMC approach.

6. CONCLUSIONS

If Conventional GMM is used for large population speaker identification system, it will increases the time complexity and also the computational complexity. Hence Optimized Hierarchical Mixture Clustering(OHMC) method was proposed, to reduce the training time and testing time and also the computational complexity with an increased identification speed. In future, we will analyze the performance of this method by comparing this approach with other speaker identification approaches.

7. REFERENCES

[1] H. Aronowitz and D. Burshtein,(2007) "*Efficient speaker recognition using approximated cross entropy"*, IEEE Trans. Audio, Speech, and Language Process, vol. 15, no. 7, pp. 2033-2043.

[2] Campbell, J.P, (2009)"*Forensic speaker recognition*", Signal Processing Magazine, IEEE 26., pp 95-103.

[3] Frederic Bimbot,(2004) "*A Tutorial on Text-Independent Speaker Verification"*, EURASIP Journal on Applied Signal Processing, vol. 4, pp. 430-451.

[4] Jayanna HS, Mahadeva Prasanna SR, (2009)"*Analysis, Feature Extraction, Modeling and Testing Techniques for Speaker Recognition*", t IETE Tech Rev, pp 90-181.

[5] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn,(200) "*Speaker verification using adapted Gaussian mixture models*", Digital Signal Process, vol.10, pp. 19-41.

[6] D.A. Reynolds, (1994)"*Experimental evaluation of features for robust speaker identification*", IEEE Trans. Speech Audio Process. , vol. 2(4), pp. 639-43.

[7] D. Reynolds and R. Rose, (1995) "*Robust text-independent speaker identification using Gaussian mixture speaker models*", IEEE Trans. Signal Process., vol. 3, no. 1, pp. 72-83.

[8] A.Revathi ,Y. Venkataramani,(2009) "*Source and System Features for Text Independent Speaker Identification using Iterative Clustering Approach"*, IEEE International Conference on Signal and Image Processing Applications, pp 1-5.

[9] R. Saeidi, H. R. Sadegh Mohammadi, T. Ganchev, R. D. Rodman, (2008) "*Hierarchical Mixture Clustering And Its Application To Gmm Based Text Independent Speaker Identification*", IEEE-Internatioal Symposium on Telecommunications, pp 770-773.

[10] B. Sun, W. Liu, and Q. Zhong,(2003) "*Hierarchical speaker identification using speaker clustering"*, International Conference Natural Lang. Process. Knowledge Eng. , pp. 299-304.