Comparative study of Speaker Recognition using GMM and RBF

[1]Lanka Koundinya, [2]PPS Subhashini
[1]M.Tech, [2]Associate Professor
Department of ECE,
R.V.R. & J.C. College of Engineering, Guntur, India

Abstract : There are so many techniques for text- independent speaker recognition. However this text-independent speaker recognition is very difficult because the recognition is performed irrespective of what one he is saying. In this work to extract features of speech signal, Linear Prediction Coefficients (LPC) and Mel-Frequency Cepstral Coefficients (MFCC) are used. The extracted features are matched using GMM using Expectation Maximization (EM) algorithm and RBF using Gradient Descent Algorithm. The two systems are compared with the performance and execution time.

***IndexTerms* – Feature Extraction, LPC, MFCC, Feature Matching, GMM, RBF**

I. Introduction

In recent years, there has been a growing interest in the use of biometric characteristics as a means of recognizing or confirming a person's identity. A person's voice is considered as one of biometric identifiers, which is supposed to be intrinsic and unique to a person and should not be reproducible by anyone else. Consequently, by using the distinguishing features in an individual's voice, a speaker recognition system can provide a higher level of non-intrusive security than conventional security procedures.[1] No two individuals sound identical because their vocal tract shapes, larynx sizes, and other parts of their voice production organs are different. In addition to these physical differences, each speaker has his or her characteristic manner of speaking, including the use of a particular accent, rhythm, intonation style, pronunciation pattern, choice of vocabulary and so on. State-of-the-art speaker recognition systems use a number of these features in parallel, attempting to cover these different aspects and employing them in a complementary way to achieve more accurate recognition.[3][4]

Speaker recognition can be divided into two categories: verification and identification. Speaker verification aims to verify whether an unknown voice matches the voice of a speaker whose identity is being claimed, while the objective of speaker identification is to identify an unknown voice from a set of known voices.[5][8] Another important feature of speaker recognition systems is whether they are text-dependent or text-independent. If a person is required to use the same text in the training and recognition session, this speaker recognition system is said to be text-dependent. In text-independent speaker recognition, the test speaker doesn't have any prior knowledge about the contents of the training phase and can speak anything.

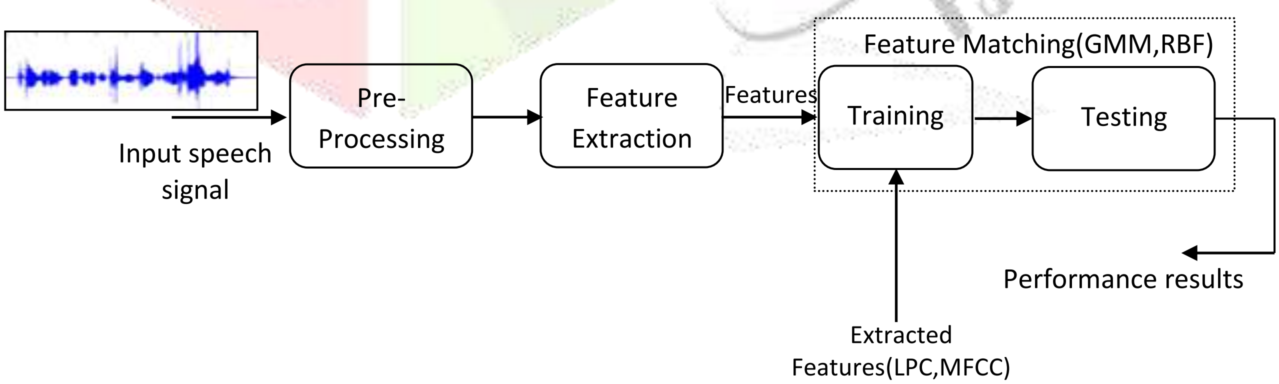

Fig. 1: Block diagram of speaker recognition

A speaker recognition system has two phases, training phase and testing phase. During training phase models are created for each speaker by extracting features from the speech signal. During testing phase, claimant features are compared with the trained speaker's feature and recognition is made by score generated from the models. The features which are used for speaker recognition are LPC(Linear Prediction Coefficients)[8] and MFCC(Mel-frequency Cepstral Coefficients)[12] which contains speaker specific information. MFCC is based on human peripheral auditory system. It is noted that human perception of frequency content of sound produced by the speaker doesn't follow linear scale. For each tone with actual frequency, there is subjective pitch measured in mel-scale. This nonlinear frequency wrapping can give better representation of voice. GMM (Gaussian Mixture Model)[1][2] is used as a parametric model for feature Classification. Maximum likelihood score is get from the mean and variances of fitted Gaussians. Accuracy of speaker recognition increases by using RBF (Radial Basis Function) for Feature Classification.

II. Feature Extraction

Feature extraction is the process that extracts a small amount of data from the speaker's voice signal that can later be used to represent the speaker. In this paper two feature extraction techniques are used, LPC and MFCC.

2.1 LPC

LPC (Linear Prediction coefficients) analyzes the speech signal by estimating the formants, removing their effects from the speech signal, and estimating the intensity and frequency of the remaining buzz. The process of removing the formants is called inverse filtering, and the remaining signal is called the residue. In LPC system, each sample of the signal is expressed as a linear combination of the previous samples. This equation is called a linear predictor and hence it is called as linear predictive coding .The coefficients of the difference equation (the prediction coefficients) characterize the formants. To decide the fundamental parameters of speech and gives exact estimation of speech parameters and computational model of discourse this LPC system is utilized. Speech test can be approximated as a direct blend of past speech tests is the fundamental thought behind LPC. The following figure2 shows the steps involved in the LPC feature extraction.

Is a reliable, accurate and robust technique for providing parameters which describe the time varying linear system which represents the vocal tract. Computation speed of LPC is good and provides with accurate parameters of speech. LPC is useful for encoding speech at low bit rate.

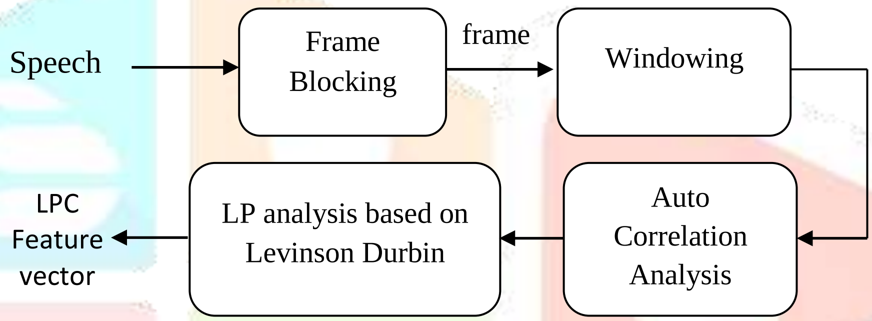

Fig. 2: Block Diagram of LPC

In this analysis first convert each frame of p+1 autocorrelations into LPC parameter set by using Durbin's method. This can formally be given as the following algorithm

$$E^{(0)} = r(0) \tag{1}$$

$$k_i = \frac{r(i)\sum_{i=1}^{l-1}\alpha_j^{l-1}r(|i-j|)}{E^{i-1}} \quad 1 \le i \le p \tag{2}$$

$$\alpha_i^{(i)} = k_i \tag{3}$$

$$\alpha_i^{(i)} = \alpha_j^{(i-1)} - k_i\alpha_{i-j}^{(i-1)} \quad 1 \le j \le i-1 \tag{4}$$

$$E^i = (1-k_i^2)E^{(i-1)} \tag{5}$$

By solving above equations recursively for *i=1,2,.....p*, the LPC coefficient m is given as

$$a_m = \alpha_m^{(p)} \tag{6}$$

these LPC coefficients are further statistically analyzed.

2.2 MFCC

These features can be obtained from the spectrogram of the speech signal and we are using Mel-Frequency Cepstral Coefficients (MFCC) features in speaker identification, the advantages of perceptual frequency scale based critical bands with cepstrum analysis are combined. On basis of literature survey MFCC is most accurate, popular and perhaps the best unknown.

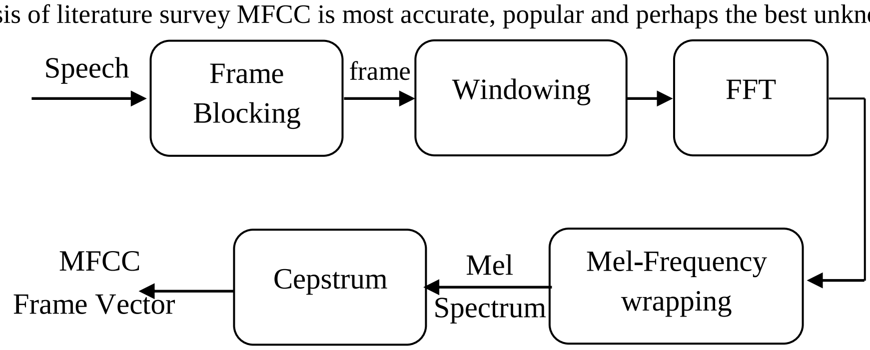

Fig. 3: Block Diagram of MFCC

The Mel frequency scale is logarithmic spacing above 1000Hz and linear frequency spacing below 1000Hz.In order to capture the phonetically important characteristics of speech frequency filters are spaced logarithmically at high frequencies and linearly at low frequencies in accordance to properties of human ear. Thus the human ear perception is clearly mimicked by MFCC. This shortly describes the process of feature extraction. Normally the speech signal is non-stationary but can be assumed as stationary for a small tenure of time, so analysis is done by framing the speech signal; the frame width is about 20−30 milliseconds, and the frames are shifted by about 10 milliseconds.

MFCC Process includes the steps hierarchically as shown in figure3. Framing is initially applied to the speech signal of the speaker partitioning the signal into N frames (segments) . In order to reduce the signal discontinuities at the start and end of each segment, the next step that is windowing is undertaken.

Later the windowed frames are processed by Fast Fourier Transform (FFT) converting frames of N samples in time domain to frequency domain. Obtained spectrum is later wrapped and converting the frequency spectrum to Mel spectrum .And finally the log Mel spectrum is converted back to time resulting in Mel Frequency Cepstrum Coefficients(MFCC). Mel scale follows the relation for an arrangement of frequency range to mel scale.

$$X_k = \sum_{i=0}^{N-1} x_i . e^{-j\frac{2\pi ki}{N}} \tag{7}$$

$$f_{mel} = 2595 log_{10}{}^{\left(1+\frac{f_{Hz}}{700}\right)} \tag{8}$$

Frame cepstrum is achieved by logarithm of amplitude of mel spectrum and applying reverse Fourier conversion:

$$mel - cepstrum(frame) = FFT^{-1}[mel(\log|FFT(frame)|)] \tag{9}$$

By taking the IFFT of the log magnitude spectrum of speech signal the FFT-base cepstral coefficients are computed. The mel-warped cepstrum is obtained by inserting a intermediary step of transforming the frequency scale to place less prominence on higher frequencies before taking the IFFT.

III. SPEAKER MODELING

In scientific field and engineering the need of speaker recognition is a much broader topic so called feature matching. Classifying the objects into number of classes and categories is the actual target of feature matching. By using the techniques, from sequences of acoustic vectors the patterns that are basically objects of interest are classified. An unknown speaker is the one with minimum matching score. For classification, speaker modeling techniques like Gaussian Mixture Model(GMM) and Radial Basis Function Neural Network are used.

3.1 GMM

GMM is a density estimator and is one of the most commonly used types of classifier for speaker recognition. In GMM, the extracted feature vector is modeled clearly using a mixture of *M* Gaussians. From a collection of training feature vectors using iterative expectation-maximization(EM) algorithm maximum likelihood model parameters can be estimated. The EM algorithm iteratively improves the GMM parameters to frequently enhance the likelihood of the estimated model for the observed feature vectors. Generally GMM is weighted sum of M component Gaussian densities and is given as

$$p(\vec{x}/\lambda) = \sum_{i=1}^{M} \omega_i g_i(\vec{x}/\mu_i, \Sigma_i) \tag{10}$$

$$g_i(\vec{x}/\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{\frac{T}{2}} \|\Sigma_i\|^{\frac{1}{2}}} exp\left\{-\frac{1}{2}(\vec{x} - \vec{\mu_i})' \Sigma^{-1} (\vec{x} - \vec{\mu_i})\right\} \tag{11}$$

The complete GMM is calculated the mean vectors μ_i, covariance matrices Σ_i and mixture weights ω_i from all components densities. These parameters are collectively represented by the notation,

$$\lambda = \{\omega_i, \vec{\mu_i}, \Sigma_i\}\ i = 1,2, \dots \dots M \tag{12}$$

The GMM has its own advantages , it is more economical and is based on a comprehensive statistical model in case of text-independent speaker verification. The shortcoming of using it is that it requires more data to model speaker parameters. It works very excellently in case of adequate data.

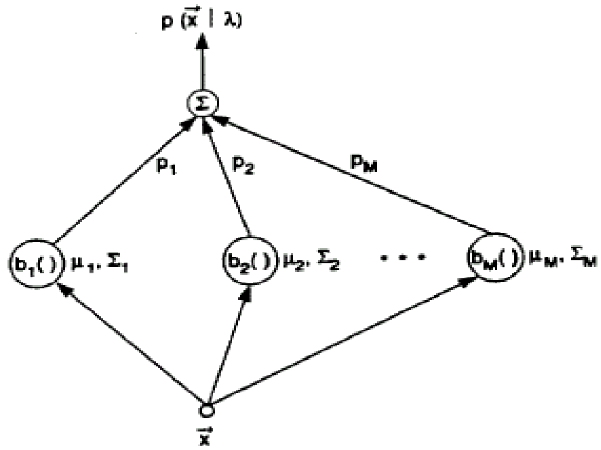

Fig. 4: Speaker Recognition using GMM

3.1.1 Expectation-Maximization Algorithm

Given a Gaussian mixture model, the goal is to maximize the likelihood function with respect to the parameters using EM algorithm.

1. Initialize the means μ_i, covariance $\sum_i$ and mixing coefficients ω_i, and evaluate the initial value of the log likelihood.
2. **E step:** Evaluate the responsibilities $\tau(z_{it})$ using the current parameter values

$$\tau(z_{it}) = \frac{\omega_i g(x_t|\mu_i, \sum_i)}{\sum_{j=1}^{M} \omega_j g(x_t|\mu_j, \sum_j)} \tag{13}$$

3. **M step**: Re-estimate the parameters μ_i^{new}, $\sum_i^{new}$, ω_i^{new} and using the current responsibilities.

$$\omega_i^{new} = \frac{T_i}{T} \tag{14}$$

$$\mu_i^{new} = \frac{\sum_{i=1}^{M} \tau(z_{it}) x_t}{T_i} \tag{15}$$

$$\sum_i^{new} = \frac{1}{T_i} \sum_{i=1}^{M} \tau(z_{it})(x_i - \mu_i^{new})(x_i - \mu_i^{new})' \tag{16}$$

4. Evaluate the log likelihood and check for convergence of either the parameters or the log likelihood. If the convergence criterion is not satisfied return to step 2.

3.2 RBF

Radial Basis Functions (RBF) are variant of feed-forward artificial neural network, that consists of at least three layers of neurons: an input layer, hidden layer and an output layer, where each hidden unit implements a radial activated function. The input into an RBF network is nonlinear while the output is linear. An RBF network with *D* inputs, *M* hidden units and *K* outputs is shown in Figure5. The output layer forms a linear combiner which calculates the weighted sum of the outputs of the hidden units.

The *k* output of an RBF neural network has the form:

$$f_k(Y) = w_{0k} + \sum_{j=1}^{M} w_{jk} \phi_j(Y) \tag{17}$$

$j=1,2,3,\ldots,M$ and $k=1,2,\ldots,K$ where w_{jk} - weights of the network. For an RBF network the activation function is:

$$\phi_j(Y) = exp\left\{-\frac{1}{2.\sigma_j^2}\left\|Y - c_j\right\|^2\right\} \tag{18}$$

where ‖.‖ denotes the Euclidean distance. In Φ_j is activation function, $Y = \{y_t, t = 1, \ldots, T\}$ is the input vector of length T and dimension D, c_j - function centers, σ_j - the function width.

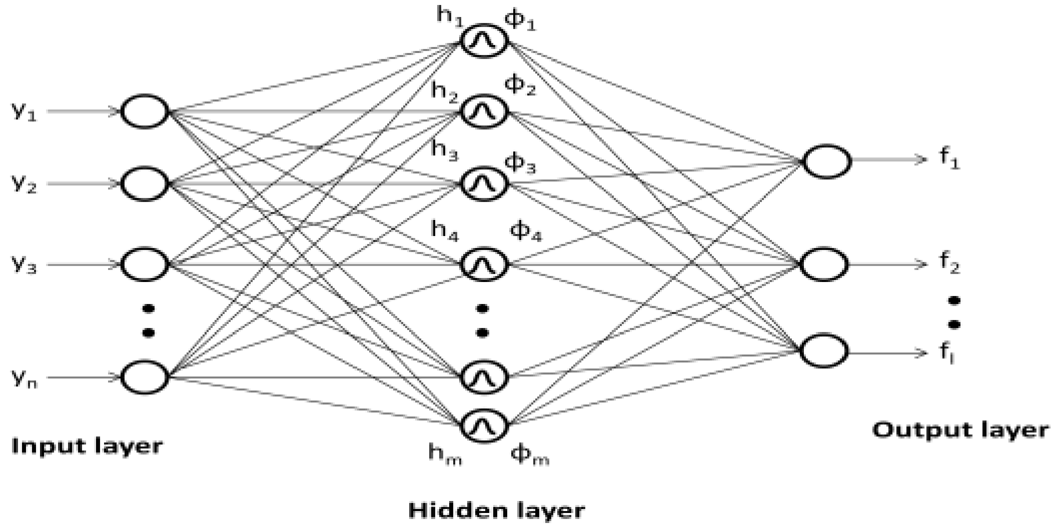

Fig. 5: Radial Basis Function Neural Network

By means of training, the neural network models the underlying function of a certain mapping. The hidden layer neurons represent a series of centres in the input data space. Each of these centres has an activation function, typically Gaussian. The activation depends on the distance between the presented input vector and the centre. The further the vector is from the centre, the lower is the activation and vice versa. The generation of the centres and their widths is done using an unsupervised k-means, clustering algorithm. The centres and widths created by this algorithm then form the weights and biases of the hidden layer, which remain unchanged once the clustering has been done. RBF networks has both a supervised and unsupervised component to its learning, but they are used mainly in supervised applications. Fully supervised training to find neuron centers, widths, and amplitude. In a supervised application, used a training set of data samples for which the corresponding network outputs are known. In this case the network parameters are found such that they minimize a cost function.

3.2.1 Gradient Descent Algorithm

Centers of GMM are proposed to determine using RBF network. Centers of RBF and other parameters of network undergo a supervised learning process. The most convenient for RBF network learning is a gradient descent algorithm that represents a generalization of the Least Mean Square (LMS) algorithm. The family of RBF networks is broad enough to uniformly approximate any continuous function on a compact set and consists of functions represented by

$$F(x) = \sum_{i=1}^{m} a_i\, \varphi(w_i^T x) \tag{19}$$

where m - the number of neurons in the first layer, a_i, w_i - coefficients of neural network, $\varphi(.)$ - activation function. As the activation function in the expression a family of exponential distributions with the shape parameter α is proposed. Calculating the mean square error of approximation of the mixture of multidimensional sampling distributions:

$$\varepsilon = \frac{1}{2}\sum_{j=1}^{N} e_j^2 \tag{20}$$

where N is the size of the training sample. Error signal defined by:

$$e_j = d_j - \sum_{i=1}^{M} w_i f(x_j - m_i) \tag{21}$$

where d_j - data. Neural network training procedure is performed incrementally using gradient descent algorithm.

Changing weights on the next step:

$$w_i(n+1) = w_i(n) - \eta_1 \frac{\partial \varepsilon(n)}{\partial w_i(n)}, i = 1, \dots, m_i \tag{22}$$

$$\frac{\partial \varepsilon(n)}{\partial w_i(n)} = \sum_{j=1}^{N} e_j(n) f(x_j - m_i(n)) \tag{23}$$

Adjustment of the position of the centers:

$$t_i(n+1) = t_i(n) - \eta_2 \frac{\partial \varepsilon(n)}{\partial t_i(n)} \;, i = 1, \ldots, m_i \tag{24}$$

$$\frac{\partial \varepsilon(n)}{\partial t_i(n)} = \alpha \omega_i(n) \sum_{j=1}^{N} e_j(n) f'\left(x_j - m_i(n)\right) \times \Sigma^{-1}(x_j - t_i(n))^{\alpha-1} \tag{25}$$

Adjustment of distribution width:

$$\Sigma_i^{-1}(n+1) = \Sigma_i^{-1}(n) - \eta_3 \frac{\partial \varepsilon(n)}{\partial {\Sigma_i}^{-1}(n)} \;, i = 1, \ldots, m_i \tag{26}$$

$$\frac{\partial \varepsilon(n)}{\partial {\Sigma_i}^{-1}(n)} = -\alpha w_i(n) \sum_{j=1}^{n} e_j(n) f'\left(x_j - m_i(n)\right) Q_{ij}(n) \tag{27}$$

$$Q_{ij}(n) = \left(x_j - m_i(n)\right)^{\alpha-1} (x_j - m_i(n))^{T} \tag{28}$$

Adjustment of the PDF shape parameter:

$$\alpha_i(n+1) = \alpha_i(n) - \eta_2 \frac{\partial \varepsilon(n)}{\partial \alpha_i(n)} \;, i = 1, \ldots, m_i \tag{29}$$

$$\frac{\partial \varepsilon(n)}{\partial \alpha_i(n)} = 2\omega_i(n) \sum_{j=1}^{N} e_j(n) f\left(x_j - m_i(n)\right) \alpha^{-1} + f'\left(x_j - m_i(n)\right)\left(\alpha \left|\frac{x - m(n)}{\lambda \Sigma}\right|^{\alpha-1}\right) \tag{30}$$

IV. Results and Discussion

The Database used is TIMIT Database. Each speaker has 10 samples which makes a total of 500 samples. Each sentence is of 3seconds. Concatenation of 8 samples to form training duration of 24 seconds and test duration of 6 seconds. Evaluation for varying mixture components is conducted for 50 speakers.

In Feature Extraction, for each frame 12 and 22 LPC coefficients are extracted. In MFCC 13 and 20 dimensional feature vector is extracted for each frame of a speech sample. In GMM, the speakers were modeled for Gaussian mixture of 2,4,8,16, 32 and 64.The execution time is calculated for training and testing models. In RBF, the network is evaluated for different number of hidden nodes between 250 and 450.

Table1. Performance analysis using GMM-EM

Features	**% of correct classification using various mixture models**					
	2	**4**	**8**	**16**	**32**	**64**
LPC-12	52.78	59.37	69.46	79.34	**85.15**	80.49
LPC-22	67.29	75.93	85.47	85.65	**88.48**	83.04
MFCC-13	71	88.3	90.8	92.4	**93.85**	88.60
MFCC-20	80.23	91.96	92.14	93.79	**94.6**	90

Table1 shows the performance analysis using GMM-EM algorithm, most of the experimental results gave 32 gaussian mixture model good performance. Table2 gives the performance analysis calculated using RBF network. For various number of Hidden nodes calculated, the best performance is obtained at 430-440 nodes.

Table2. Performance analysis using RBF

Features	% of correct classification using various No. of Hidden nodes						
	355	370	380	410	425	435	450
LPC-12	64.36	66.82	68.49	73.45	80.61	90.23	**91.26**
LPC-22	70.45	73.56	78.19	83.81	86.65	89.53	**92.54**
MFCC-13	76.49	80.25	83.56	92.49	93.56	94.12	**95.49**
MFCC-20	84.59	85.96	89.45	90.27	92.43	95.01	**96.15**

Table3. Execution Time for training and testing using GMM

Features	Feature matching	Execution time for training and testing using various mixture models					
		2	4	8	16	32	64
LPC-12	Train	21.916	41.620	61.860	54.563	141.36	259.13
	Test	49.179	51.833	54.563	59.314	57.673	55.702
LPC-22	Train	31.816	65.235	89.426	150.81	405.60	898.33
	Test	51.589	51.257	55.210	57.827	64.087	81.392
MFCC-13	Train	24.131	41.976	51.946	73.983	120.96	242.78
	Test	48.232	52.734	52.451	56.674	51.074	55.269
MFCC-20	Train	24.934	31.719	62.136	87.992	159.60	398.19
	Test	54.254	51.481	50.337	56.228	50.208	57.009

Table3 gives the execution time for training and testing of the GMM. In testing for any mixture model, runtime is 50-60 sec approximately. In training for high mixture models take more runtime. Table4 gives the execution time for training and testing of the RBF. Results showed that the execution time for RBF is faster when compared to GMM.

Table4. Execution Time for training and testing using RBF

Features	Feature Matching	Execution time for training and testing using various No. of Hidden nodes						
		355	370	380	410	425	435	450
LPC-12	Train	80.96	83.45	100.07	120.64	130.25	141.36	159.13
	Test	27.179	26.45	26.833	24.563	29.325	24.32	26.482

LPC-22	Train	110.45	115.49	126.87	139.28	159.62	176.40	190.03
	Test	29.90	27.64	25.75	26.21	27.347	22.087	30.392
MFCC-13	Train	73.15	74.87	76.826	85.60	96.39	120.96	136.49
	Test	28.56	27.74	21.595	27.685	24.658	26.495	24.364
MFCC-20	Train	76.95	79.49	84.359	90.259	94.756	130.49	150.49
	Test	23.67	27.25	26.59	22.159	21.91	23.478	22.456

Figure6 shows the average performance of GMM and RBF with LPC-12,22 and MFCC-13,20. Results showed that the MFCC outperform LPC in all experiments. RBF gives higher performance compared to GMM in all the cases.

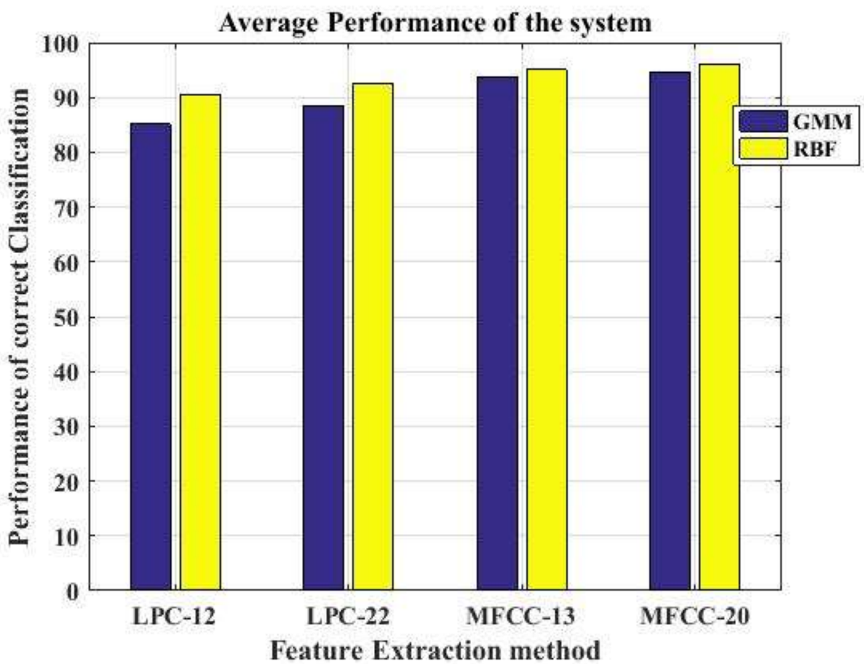

Fig. 6: Average Performance of the system

CONCLUSION

The text-independent speaker recognition is very difficult compared with the text- dependent speaker recognition because here the testing is performed with the new inputs which are not there in training. So the new methods are necessary and the present study is still on-going. Main consideration in the speaker recognition problem has been given to the selection of features. The recognition accuracy of current speaker recognition systems under controlled conditions is high. Results showed that the MFCC outperform LPC in all experiments. The main advantage of RBF over GMM is that it is unaffected by the differing shape and style of testing speech as the network is already trained with large variations. The Execution time of RBF is low compared to GMM. RBF gives higher performance compared to GMM in all the cases.

REFERENCES

[1] D. A. Reynolds,2001. Gaussian Mixture Models, Technical Report, MIT Lincoln Laboratory, Cincinnati.

[2] D. A. Reynolds, T. F. Quatieri and R. B. Dunn,2000. Speaker Verification Using Adapted Gaussian Mixture Models, Digital Signal Processing, Vol. 10, No. 2: 19- 41.

[3] Anton Yakovenko and Galina Malychina,2016. Text-Independent Speaker Recognition Using Radial Basis Function Network, Springer International Publishing Switzerland, ISNN 2016:74–81.

[4] Deng Jiamping and Ronda Venkateswarulu, Speaker Recognition Using Radial Basis Function Neural Networks,Springer-Verlag Berlin Heidelberg:57-64.

[5] Najiya Abdulrahiman,Ranju K.V.,2013. Text Dependent Speaker Recognition, International Journal of Electronics and Delta Communication,ISSN:2320-2084,1(7):05-11.

[6] Vibha Tiwari,2010. MFCC and its applications in speaker recognition, International Journal on Emerging Technologies, ISSN : 0975-8364 1(1):19-22.

[7] Sonali T. Saste,Prof. S. M. Jagdale,2017. Comparative Study of Different Techniques in Speaker Recognition: Review, International Journal of Advanced Eng.,Management and Science,ISSN: 2454-1311,3(3):284-287.

[8] Jose B. Trangol Curipe and Abel Herrera Camacho,2013. Feature Extraction Using LPC-Residual and Mel Frequency Cepstral Coefficients in Forensic Speaker Recognition, International Journal of Computer and Electrical Engineering, 5(1):40-43.

[9] J. G. Proakis and D. G. Manolakis,2009. Digital Signal Pro- cessing, Prentice Hall, Boston.

[10] D. O'Shaughnessy,1987.Speech Communication: Human and Machine, Addison-Wesley, Boston.

[11] R. Reynolds,19945. Robust Text-Independent Speaker Identi-fication Using Gaussian Mixture Speaker Models, IEEE Transactions on Speech and Audio Processing:72-83.

[12] Jayanthi kumari T.R, H. S. Jayanna,2014. Comparison of LPCC and MFCC features and GMM and GMM-UBM modeling for Limited Data Speaker Verification, IEEE International Conference on Computational Intelligence and Computing Research.

[13] Bishu Prasad Das,Ranjan Parekh,2012. Recognition of Isolated Words using Features based on LPC,MFCC,ZCR and STE, with Neural Network Classifiers, International Journal of Modern Engineering Research(IJMER),2(3):854-858.

[14] "TIMIT Speech Database." http://www.ldc.upenn.edu

[15] Aliyu E.O.,Adewale,O.S.Adetunmbi A.O.,2013. Development of a Text-Dependent Speaker-Dependent Speaker Recognition System, International Journal of Computer Applications,69(16),01-07.

[16] S.B.Dhonde, S.M.Jagade,2015. Mel-Frequency Cepstral Coefficients for Speaker Recognition : A Review, International Journal of Advance Engineering and Research Development,2(5):1115-1119.

[17] R. B. ShindeÅ and V. P. Pawar,2013. Fusion of MFCC & LPC Feature Sets for Accurate Speaker Identification, International Journal of Current Engineering and Technology, ISSN 2277 – 4106:1763-1766.

[18] Kshamamayee Dash, Debananda Padhi , Bhoomika Panda, Prof. Sanghamitra Mohanty,2012. Speaker Identification using Mel Frequency Cepstral Coefficient and BPNN, International Journal of Advanced Research in Computer Science and Software Engineering, ISSN: 2277 128X, 2(4):326-332.

[19] Shreya Narang1, Ms. Divya Gupta,2015. Speech Feature Extraction Techniques: A Review, International Journal of Computer Science and Mobile Computing, ISSN 2320–088X ,4(3):107-114.

[20] Jyoti B. Ramgire, Prof. Sumati M.Jagdale,2016. A Survey on Speaker Recognition With Various Feature Extraction And Classification Techniques, International Research Journal of Engineering and Technology (IRJET),ISSN: 2395 -0056, 03.