LFBNN: Robust and Hybrid Training Algorithm to Neural Network for Hybrid Features-Enabled Speaker Recognition System

Vasamsetti Srinivas* and Ch. Santhirani**

**Acharya Nagarjuna University College of Engineering, Guntur, A.P-522510, India*
***Professor & Dean Academics, Usha Rama College of Engineering & Technology, Gannavaram, A.P-521101, India*
**Corresponding Author: srinivas.siet@gmail.com*

Submitted: 01/08/2018
Revised: 06/10/2018
Accepted: 08/10/2018

ABSTRACT

The field of speaker recognition found a major challenge among the research community due to the lack of robustness of earlier methods against environmental noises. To effectively handle the robustness in the speaker recognition method, this paper aims to develop a new classification algorithm for automatic speaker recognition. This paper introduces a method called, Levenberg- Fractional Bat Neural Network (LFBNN) with hybrid features for robust speaker recognition. For feature extraction, four different features, that is, autocorrelation coefficients, predictivity ratio, MFCC, and spectral centroid, are utilized to construct the feature library. Then, the feature library is utilized to train the feed forward neural network using the proposed Fractional Bat Levenberg Marquardt (FBLM) algorithm. FBLM is designed by integrating the Fractional Bat Algorithm and Levenberg Marquardt Algorithm (LMA), for training the feed forward neural network. The experimentation is performed with the ELSDSR database, and the performance is evaluated with a few existing methods using three different evaluation metrics, like FAR, FRR, and accuracy. From the robustness analysis, it is proved that the accuracy of the proposed method is 95 %.

Keywords: Speaker recognition; Speech signal; Levenberg- Fractional Bat Neural Network (LFBNN); Optimization; Accuracy.

1. INTRODUCTION

Biometric recognition systems are utilized as the most natural way of people recognition. As an alternative to remembering passwords and PINs, which can be forgotten or stolen, biometric indications, like face, voice, and fingerprints, are peculiar to an individual and indicate that person (Jain *et al*., 2004). Speaker recognition systems (Avci, 2009; Reynolds *et al.,* 2000; Wu & Lin, 2009) utilize the human speech for recognizing, identifying, or verifying an individual. Speaker recognition (Nath & Kalita, 2014; Moeikham & Srinonchat, 2008) is a special classification task, which has a large number of classes and each class has a minimum number of samples. Speaker recognition is considered as a solution to the classification problem, where a pair of patterns demonstrating an unknown speaker utterance is obtained to validate the hypothesis that the two utterances belong to the same speaker (Cumani & Laface, 2014).

Speaker recognition is the process of identifying a person from voice characteristics. There are two main applications of speaker recognition methods, that is, speaker identification and speaker verification. Speaker identification is the process of finding who is talking from a set of unknown voices of speakers. Speaker identification determines who has provided a given speech depending on the information presented in speech waves. It is also called closed set identification, since the unknown voice comes from a set of known speakers. It is a 1: N match, in which the voice is compared against N templates. Speaker Verification is the process of declining or accepting the speaker claim to be the actual one. It is a 1:1 match, in which the voice of one speaker is matched to a voice model. The errors in speaker verification are divided into two groups, false rejections and false acceptances. In false rejections, a true speaker is discarded as a pretender and in false acceptances, a false speaker is accepted as a true speaker (Saquib et al., 2010). Training and testing are two operational phases in speaker recognition. In the training phase, the speech of each speaker is attained for training the model for speakers. The model is created for the speaker as an off-line process during the time of system configuration. In the testing phase, the speech from an unidentified utterance is compared with every trained speaker models (May *et al.,* 2012).

Large numbers of mismatches happen among the training and testing conditions because of several causes, as vocal effort, emotions, language, accent, communication channel, the resonance of room, background noise, and so on (Sadjadi & Hansen, 2014). This results in a major challenge in providing robustness to the speaker recognition system (Sadjadi & Hansen, 2012; Togneri & Pullella, 2011; Jain *et al.,* 2004; Ming *et al.,* 2007; Kinnunen & Li, 2010; Campbell *et al.,* 2009; Campbell *et al.,* 2006; Pullella *et al.,* 2009). The research community is providing attention in the direction of the challenge in designing a vigorous speaker identification system. For compensating the effect of environmental noise, the recognizer in the noise-robust speaker recognition is modified in such a way that it considers only the feature components that have the consistent information regarding the target signal.

In the proposed method, LFBNN is developed by combining LMA and Fractional Bat Algorithm (Thomas & Rangachar, 2017) with hybrid features for robust speaker recognition. The proposed system contains the two major steps, that is, feature extraction and classification. Initially, the training speech signals of various speakers are obtained for feature extraction steps. In feature extraction, two categories of features, that is, time and frequency domains, are extracted. Here, time domain features, like autocorrelation coefficients and Predictivity ratio, and frequency domain features, like Mel-Frequency Cepstral Coefficients (MFCC) and Spectral centroid, are utilized. These two domains of features are integrated to preserve the uniqueness of every speaker hidden in both domains of features. In the classification phase, the neural network is trained with the features extracted using the proposed FBLM training algorithm. FBLM is developed by integrating the Levenberg–Marquardt algorithm and Fractional Bat Algorithm. In the testing phase of the neural network, LFBNN tests the features of the input signal, and it recognizes the person based on the trained model.

The main contributions proposed in the paper are given as follows:

- Hybrid features are used to extort the unique characteristics of the speakers using four different features, that is, autocorrelation coefficients, predictivity ratio, MFCC, and Spectral centroid.
- An LFBNN classifier model is introduced by integrating a neural network model with the FBLM training algorithm. Here, FBLM is designed by combining Fractional Bat Algorithm and LMA. Fractional Bat Algorithm is the optimization algorithm that is comprised of Bat algorithm and Fractional theory. LMA is also a kind of gradient algorithm developed for searching purpose.

The organization of the paper is as follows: Section 2 presents the literature review. Section 3 explains the problem statement. Section 4 presents the proposed speaker recognition method. Section 5 experiments the proposed method and discusses the performance improvement. Finally, the conclusion is presented in Section 6.

2. LITERATURE REVIEW

This section shows the review of different speaker recognition methods availed recently in the literature. Here, Gaussian Mixture Model (GMM) and neural network are utilized by most of the researchers for speaker recognition. The major disadvantage observed in these methods is that the GMMs are not adaptive against the noisy information. Also, feature extraction utilized by these methods is a traditional one, but the unique characteristics of different features available in the literature can provide improved performance than a single feature extraction method. Richard McClanahan and Phillip L. De Leon (Clanahan & Leon, 2015) developed a multi-layered hash system using a tree-structured Gaussian Mixture Model-Universal Background Mode (GMM–UBM) that adopted Runnalls' Gaussian mixture reduction method to reduce the computational complexities regarding the calculations of posterior probabilities and statistics. The complexity in the calculation is reduced using GMM–UBM, but the hash GMM requires several kilo-bytes of storage additionally. S. Cumani and P. Laface (Cumani & Laface, 2018) have introduced a speaker modeling method, named as "e–vector," which extracts a compact representation of a speech segment, like the speaker factors of Joint Factor Analysis (JFA) and i–vectors. This method provided the best performance with no extra computational or memory costs, but it is not helpful for Pairwise SVM (PSVM).

L. Xu *et al.* (2018) have presented a method, named Rapid subspace-orthogonalizing-prior (Rapid SOP), for i-vector extraction for speaker recognition. In this method, the full posterior covariance was not evaluated. Hence, the run-time extraction process was speeded up. Here, the computational demand was high for large senone set. Md Sahidullah and Goutam Saha (Sahidullah & Saha, 2013) presented a windowing approach to calculate MFCC for automatic speaker recognition. The technique depends on the basic property of Discrete-Time Fourier Transform (DTFT). As the order of the window was increased, the spectral leakage was also increased with reduced sidelobe attenuation affecting the performance of recognition. An approach was designed for speaker recognition by Sandro Cumani and Pietro Laface (Cumani & Laface, 2014) to train a Pairwise Support Vector Machine (PSVM) using an appropriate kernel for i–vector pairs. This approach allowed rejecting unnecessary training pairs without affecting the accuracy. Although memory is required and computational computations are reduced, the approach is slow for the large training set. To suppress the reverberation overlap-masking effect on Speaker Identification (SID) systems, Seyed Omid Sadjadi and John H. L. Hansen (Sadjadi & Hansen, 2014) presented a technique called Blind Spectral Weighting (BSW). The performance of SID is increased, but the verification performance is degraded due to increasing reverberation time.

P. Alku and R. Saeidi (Alku & Saeidi, 2017) have introduced a linear predictive spectral estimation technique, called combined higher-lag linear prediction (CHLLP), that depends on higher-lag autocorrelation coefficients for the noise-robust feature extraction from speech. CHLLP offered the high performance, but it had a larger value of error. O. Ghahabi and J. Hernando (Ghahabi & Hernando, 2017) presented an impostor selection algorithm and a universal model adaptation process in a hybrid system that depends on deep neural networks and deep belief networks to model the target speaker. This method filled the performance gap in the decision cost function. However, the performance of this method was lower than the baseline PLDA system. S. Ranjan and J. H. L. Hansen (Ranjan & Hansen, 2018) proposed a class of curriculum learning (CL) based algorithms for noise robust speaker recognition. CL-based methods were included at two stages, (i) i-Vector extractor estimation and (ii) probabilistic linear discriminant (PLDA) back-end. The CL-based PLDA provided considerable improvements than the conventional PLDA based back-end.

Ehsan Variani *et al.* (Variani et al., 2014) analyzed the utilization of Deep Neural Networks (DNNs) for a small footprint text-dependent speaker verification task. Initially, a DNN is trained to classify the speakers at the frame level and then, the trained DNNs extract the features of the speaker from the last hidden layer. After that, the average of the extracted features (d-vector) is considered as the speaker model. Finally, at the evaluation, the features are extracted for every speech and compared to the registered speaker model for taking a verification decision. The DNN based system is vigorous to additive noise and outperforms the i-vector system. Amirsina Torfi *et al.* (Torfi et al., 2017) introduced a technique for speaker verification in the text-independent setting using 3D Convolutional Neural Network (3D-CNN). This technique captures the speaker-related information and builds a vigorous system concurrently. This technique outperforms the d-vector verification system. The drawback of this technique is that it did not handle the several numbers of speaker utterances.

Siyang Song *et al.* (Song et al., 2018) developed a pre-processing scheme named Noise Invariant Frame Selection (NIFS). This method chooses the noise invariant frames from speeches and depends on the various noisy constraints, for representing speakers. This method is robust and simple to reproduce. The NIFS is valuable for both testing and training, and it is appropriate for various features. Chunlei Zhang *et al.* (Zhang et al., 2017) investigated the utilization of deep learning methods for spoofing detection in speaker verification. Depending on the analysis, a spoofing detection system is developed, which employs CNN and Recurrent Neural Network (RNN). Here, CNN is used as a convolutional feature extractor, and RNN is employed for capturing long-term dependencies across the time domain. This system reimburses for the problem due to short duration test utterances.

3. PROBLEM STATEMENT

Recognition or identification of a person through speech signal is the challenging task in speaker recognition. The major challenge in speaker recognition is caused by the inconsistencies in the types of audio and their quality. Speech recognition systems are separated in various classes by describing the type of vocabulary, type of channel, type of speaker model, and the type of speech utterance. Speech recognition is a complex and challenging task due to this variability in the signal. The vocabulary size of a speech recognition system influences the accuracy, processing requirements, and the complexity of the system. Let us assume D as the input database having U speakers and every speaker u_i has Z training speech signals.

$$D \in u_i \ ; \ \{1 \le i \le U\} \tag{1}$$

$$u_i = \left\{ z_j \quad ; \quad 1 \le j \le Z \right\} \tag{2}$$

Every speech signal corresponding to z_j is denoted as a signal vector $s(i)$, which is utilized to recognize the speaker. The main problem considered here is to recognize U speakers separately based on their speech signal.

Despite speaker recognition, the real-time issue is in performing person recognition through their speech signal even if the input signal is noisy. While taking the input speech signal from the human, the chance to get the information affected with noise is high, due to environmental facts. Hence, noisy environmental conditions are important factors to be considered by the researchers to develop a noise-adaptive classification algorithm.

In Sahidullah & Saha (2013), MFCC is utilized for speaker recognition, whereas GMM is employed in Clanahan & Leon (2015). MFCC fails to identify the significant features from the speech signal if the amount of noisy information is large and GMM requires distribution characteristics of the input signals for classification. Moreover, GMM does not fit for noisy environmental conditions (Clanahan & Leon, 2015).

4. PROPOSED ROBUST AND HYBRID TRAINING ALGORITHM TO NEURAL NETWORK FOR HYBRID FEATURES-ENABLED SPEAKER RECOGNITION SYSTEM

The ultimate objective of this paper is to design and develop an automatic speaker recognition system. Here, a method called LFBNN is developed with hybrid features for robust speaker recognition. Figure 1 shows the block diagram of the proposed system. The proposed systems contain the two major steps, namely, feature extraction and classification. Initially, the speech signals of the speakers are provided to the feature extraction steps. In this step, hybrid features dependent on time and frequency are extracted for every signal to construct the feature library. Here, time domain features, such as Autocorrelation Coefficients and Predictivity ratio, are utilized. For frequency domain, MFCC and Spectral centroid are utilized. Finally, FFNN (Pan *et al.,* 2005; El-Melegy, 2013) is trained based on the proposed training algorithm, which is developed by combining the LMA (Pujol, 2007) and FBA (Thomas & Rangachar, 2017).

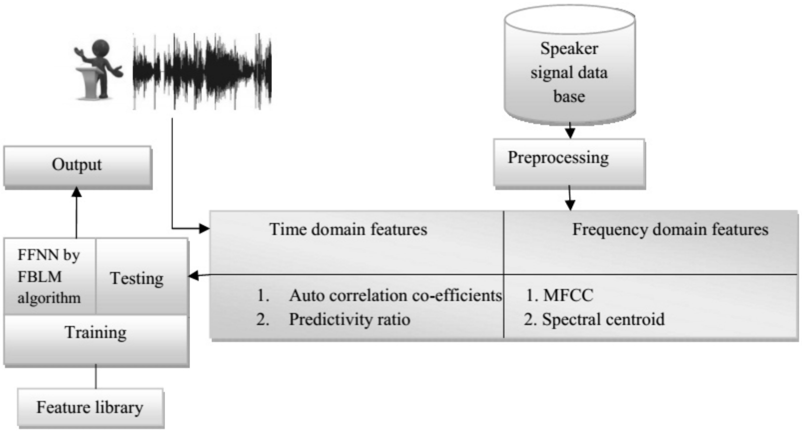


Figure 1. Block diagram of the proposed system.

4.1 Feature extraction

Here, the unique characteristics of speakers hidden in their speech signal are identified. In speech processing, two types of the feature, time domain features and frequency domain features, are extracted depending on the application and requirement. Time domain features extract the original characteristic of the signal, whereas, in the frequency domain analysis, the hidden unique characteristics are extracted through the transformation. Here, two domains of the feature are integrated to preserve the uniqueness of every speaker hidden in both domains of features. Time domain features, like autocorrelation coefficients and Predictivity ratio, are extracted to find out the unique characteristics of speakers. Also, frequency domain features, such as MFCC and Spectral centroid, are utilized.

a) Autocorrelation coefficients

Autocorrelation coefficients (Morales-Cordovilla *et al.,* 2011) are used for real-time signal processing to find the correlative behavior at various time intervals. This symmetric analysis of speakers signal over a time period can give some unique characteristics, which can efficiently differentiate the speakers. The extraction of autocorrelation coefficients $rr_s(p)$ from $s(i)$ is represented as follows:

$$rr_s(p) = \frac{1}{N}\sum_{i=p}^{N-1} s(i)s(i-p) \quad \left(0 \le p \le N\right) \tag{3}$$

where p represents the delay in samples and N represents the total number of samples in the input speech signal.

b) Predictivity ratio

Predictivity ratio (Burred & Lerch, 2003) refers to the cumulative ratio of the energy of the predicted sample to the original signal over a period of time. This feature contributes more to the uniqueness of every speaker through the predicted coefficients, which consider multiple previous samples. As identification of unique characteristics is essential in speaker recognition, predictivity ratio is measured. The definition of computing the predictivity ratio $P_R(p)$ using the delay samples p is given as follows:

$$P_R(p) = \frac{\sum_{i=0}^{N} \left|\hat{s}(i)\right|^2}{\sum_{i=0}^{N} \left|s(i)\right|^2} \tag{4}$$

where $\hat{s}(i)$ is the prediction of its amplitude value as a linear combination of its past p samples. $s(i)$ is the original signal sample. The prediction of the signal $\hat{s}(i)$ is represented as

$$\hat{s}(i) = a_1 s(i-1) + a_2 s(i-2) + \ldots + a_p s(i-p) \quad (0 \le p \le N) \tag{5}$$

where a_i is called Linear Prediction Coefficients (LPC), which is obtained by the autoregressive modeling. The advantage of LPC coefficients (Fischer, 2013) is that it is adaptable to changes and high noise components even if the signal is noisy.

c) Mel Frequency Cepstral Coefficient (MFCC)

Mel Frequency Cepstral Coefficient (On *et al.,* 2006) is the traditional and effective methods widely applied for feature extraction in speech processing. The major advantage is that it can easily understand the speech signal with accurate representation of the phoneme and envelope of the spectrum. These characteristics can contribute more to the recognition of speakers. The steps to be followed for extracting MFCC from the input speech signal are given as follows:

1) At first, the input sample of the signal, $s(i)$, is divided into a set of frames based on the length L of the frame desired and the overlapping size of the frames, o_s. After dividing the frames, frames are padded with zeros only if there are odd numbers of frames. The frames are represented as $s_l(i)$, where l ranges over 1 to L.

2) For every frame of signal sample $s_l(i)$, Discrete Fourier Transform (DFT) is found out using the following equation:

$$F_l(k) = \sum_{i=1}^{N} s_l(i) * h(n) * e^{-2*\pi*j*i*k/N} \qquad 1 \le k \le K \tag{6}$$

where $F_l(k)$ is the DFT of the frame number l, $h(n)$ is an N sample long analysis window, and K represents the length of the DFT.

3) The periodogram-based power spectral estimate for the speech frame $s_l(i)$ is represented as

$$P_l(k) = \frac{1}{N} \left| S_l(k) \right|^2 \tag{7}$$

where $P_l(k)$ is the power spectrum of the number l. This estimate is also named as periodogram estimate of the power spectrum.

4) Once periodogram estimate of power is computed, the mel-spaced filter bank is designed with m triangular filters. The designed filter bank contains $f(m)$ vectors of length Q. Here, Q denotes the Q point FFT considered. The filter bank characteristics $H_m(k)$ are given as follows:

$$H_m(k) = \left\{ \begin{matrix} 0 & k < f(m-1) \\ \dfrac{k - f(m-1)}{f(m) - f(m-1)} & f(m-1) \le k \le f(m) \\ \dfrac{f(m+1) - k}{f(m+1) - f(m)} & f(m) \le k \le f(m+1) \\ 0 & k > f(m+1) \end{matrix} \right\} \tag{8}$$

where m is the number of mel-spaced filters desired and $f()$ is the list of m+2 Mel-spaced frequencies.

5) Every filter bank is multiplied with the power spectrum and the logarithm is taken for computing the filter bank energy $E(m)$.

$$E(m) = log\left(\sum_{k=1}^{M} H_m(k)^* P_l(k)\right) \tag{9}$$

6) Once the filter bank energy $E(m)$ is found, Discrete Cosine Transform (DCT) of the m log filter bank energies is computed to obtain m cepstral coefficients.

$$C(m) = \frac{1}{2}(E_0 + (-1)^k E_{m-1} + \sum_{q=1}^{M-2} E_q\, cos\left[\frac{\pi}{m-1} qm\right] \qquad m = 0,\ldots,M-1 \tag{10}$$

where $C(m)$ is m cepstral coefficient desired.

d) Spectral centroid

The spectral centroid (Kua *et al.,* 2010) is the frequency dependent feature mainly used to measure the brightness of the sound. The spectral centroid is calculated by assessing the "center of gravity" through the magnitude information and Fourier transform's frequency. It is termed as the average frequency weighted by amplitudes and divided by the sum of the amplitudes.

$$S_C = \frac{\sum_{k=1}^{M} \tilde{k}^* F(k)}{\sum_{k=1}^{M} F(k)} \tag{11}$$

$$F(k) = \frac{1}{N} \sum_{i=0}^{N-1} s(i)^* e^{-2^*\pi^* j^* i^* k/N} \tag{12}$$

where $F(k)$ are the DFT coefficients.

e) Feature concatenation

Once these four features are extracted for a sample $s(i)$, features are concatenated into a single vector and denoted as follows:

$$r = \left[S_C ; C ; P_R ; rr_s\right] \tag{13}$$

where r is the feature extracted for a signal sample and S_C, C, P_R, rr_s are spectral centroid, MFCC, predictivity ratio, and autocorrelation coefficients. The final feature vector can be represented as

$$R = \{r_i\} \qquad ; \; 0 \leq i \leq d \tag{14}$$

where, $d = 1 + m + p + p$ (15)

where R is the feature vector and d is the size of the feature vector. The feature extraction process extracts the features, autocorrelation, predictivity ratio, MFCC coefficients, and spectral centroid, and is demonstrated in figure 2.

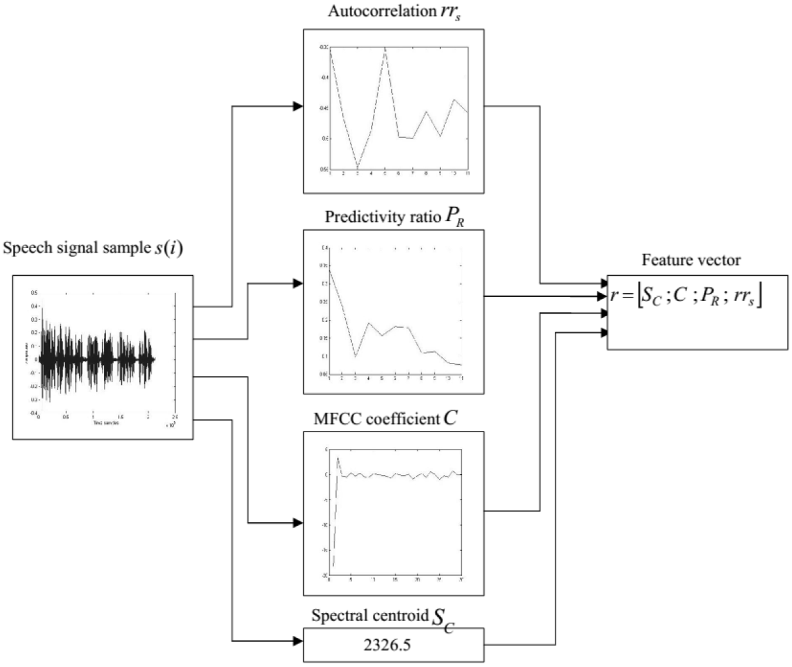


Figure 2. Feature Extraction.

4.2 Classification

Classification is the second module of this work used to recognize the speaker using the features extracted in the previous step. Even though many classification models are presented in the literature, FFNN is chosen for speaker recognition. Here, two important phases are defined, namely, i) training phase and ii) testing phase. At the training phase, *Z* training speech signals of *U* speakers are taken, and the features are extracted to build the feature library of *R*. The feature library is used to train the neural network using the proposed FBLM algorithm. FBLM is a newly developed algorithm by integrating the FBA and LMA. In the testing phase, for the test speech signal, the trained neural network classifies or identifies the speaker after the input feature vector of the test signal sample is applied to the trained neural network.

a) Feed forward neural network

The classification is done using the neural network (Pan *et al*., 2005; El-Melegy, 2013), which is the popular method used mainly for recognition. Figure 3 shows the neural network architecture.

The input for the neural network is a feature library computed from the previous step. The input elements r_j are multiplied with the node weight of the input layer like the equation given below.

$$A_1^{(j)} = r_j * w_{1j} \; ; \; 0 \le j \le d-1 \qquad (16)$$

where *d* is the number of input elements in every feature vector, i.e., the number of input neurons. The output of the input layer is given as input to the hidden layer 1. At the hidden layer of the neural networks, the output of every input nodes is added and multiplied with the bias weights a_{1l}, as represented below,

$$B_1^{(l)} = \left(\sum_{j=1}^{d} A_1^{(j)}\right) * a_{1l} \; ; \; 0 \le l \le e \qquad (17)$$

where *e* is the number of hidden neurons in the hidden layer 1. Then, to handle the linearity among the output, a sigmoid function is used in the hidden layer.

$$O_{B1}^{(l)} = \frac{1}{1+e^{-B_1(l)}} \tag{18}$$

Again, the values are multiplied with the node weight of the hidden layer, which will be added, and then multiplied with the bias weights of the output layer.

$$A_2^{(l)} = O_{B1}^{(l)} * w_{2j} \quad ; 0 \le l \le e1 \tag{19}$$

The multiplied output is processed through sigmoid function and further multiplied with the node weight of the output layer to obtain the final output in the output layer.

$$B_2^{(l)} = \left(\sum_{l=1}^{e1} A_2^{(l)}\right) * a_{2l} \quad ; 0 \le l \le e1 \tag{20}$$

$$O_{B2}^{(l)} = \frac{1}{1+e^{-B_2(l)}} \tag{21}$$

$$Y(u) = O_{B2}^{(l)} * w_{3j} \quad ; 0 \le u \le U \tag{22}$$

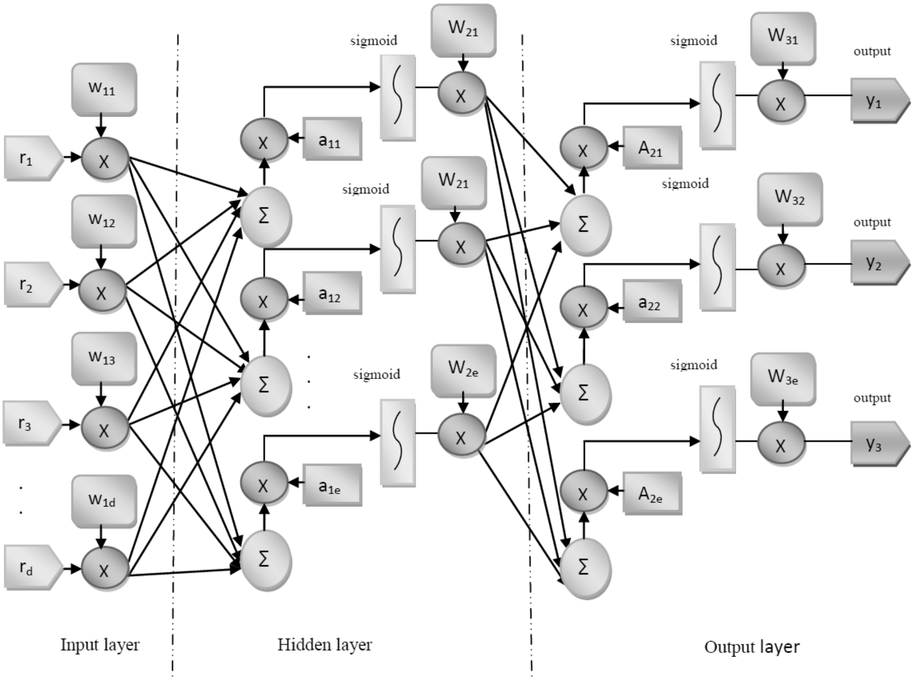


Figure 3. Neural network architecture.

b) Neural network training by the proposed FBLM algorithm

Training of neural network is a significant step in building network model. The classification performance is mainly based on the training algorithm, which is used to find the weights, such as node weights and bias weight optimally for the training data. In this paper, FBA (Thomas & Rangachar, 2017) and LMA (Pujol, 2007) are integrated for optimally finding the weights of the neural network. FBA is the integration of Bat algorithm and Fractional theory, while LMA is also a kind of gradient algorithm developed for searching purpose. Here, the weights are searched using the proposed hybrid algorithm. FBA detects any obstacles in the speech signal and it obtains the optimal weight value of the neural network. Also, FBA is robust to handle the noisy regions of the speech signal. The LMA has several advantages, such as being robust, faster to converge, and handling models with multiple free parameters. Hence, the proposed FBLM has the advantages of both FBA and LMA and finds the optimal weights of the neural network. Let us assume that the neural network utilized in the proposed system has the weight vector *W*, represented as

$$W = [\, w_{11}, w_{12}, \ldots, w_{1d}, w_{21}, w_{22}, w_{2e}, \ldots, w_{31}, w_{32}, w_{3U}, \ldots, a_{11}, a_{12} \ldots a_{1e}, a_{21}, a_{22} \ldots a_{2U} \,] \tag{23}$$

where *w* represents the node weights and *a* is bias weight.

Initially, the weights are randomly initialized and then, the weights are updated dynamically at every iteration *t*. The formula for updating the weights of each iteration using the LMA is given as follows:

$$W_{t+1}{}^{LM} = W_t - [\, H + \mu * I \,]^{-1} * g \tag{24}$$

$$H = J^T * J \tag{25}$$

where μ is the Levenberg's damping factor ranging from 0 to 1, *I* refers to the identity matrix, and *J* refers to the Jacobian matrix for the system, which is attained by considering the first-order partial derivatives of a vector-valued function. Jacobian matrix is calculated by determining the partial derivatives of every output with regard to every weight, which is represented as

$$J = \begin{bmatrix} \dfrac{\partial F(r_1, w)}{\partial w_1} & \cdots & \dfrac{\partial F(r_1, w)}{\partial w_x} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial F(r_d, w)}{\partial w_1} & & \dfrac{\partial F(r_d, w)}{\partial w_x} \end{bmatrix} \tag{26}$$

where *r* denotes the feature vector given to the neural network, *W* is the weight of the network, and *F*(*r*,*W*) denotes a nonlinear function for the neural network. *x* is the total number of weights, i.e., node and bias weight defined in the neural network.

The following equation calculates the gradient matrix of *g*.

$$g = J^T * E \tag{27}$$

Once the weights are computed, the output of the neural network is computed after applying weights to the neural network function.

$$Y^{LM} = F(\, r,\, W^{LM} \,) \tag{28}$$

where $F(r,\ W^{LM})$ denotes the network function, which is calculated for each feature vector of the training signal using the weight vector. *W* and *Y* denote the associate output vectors predicted or approximated by the network. Using the neural network output and original ground truth E^{LM} is computed based on the output errors of every input vector utilized for training the network.

$$E^{LM} = \frac{1}{N} \sum_{i=1}^{N} \left(Y^{LM}{}_i - Y_{Gi} \right) \tag{29}$$

In the current iteration, the formulae used for updating the weights of every iteration using FBA are given as follows

$$W_{t+1}^{FB} = \alpha W_i^t + \frac{1}{2}\alpha W_i^{t-1} + \frac{1}{6}\alpha(1-\alpha)W_i^{t-2} + \frac{1}{24}\alpha(1-\alpha)(2-\alpha)W_i^{t-3} + \eta L^t \tag{30}$$

where η is the random number from 0 to 1, α is the fractional derivative order with its value in the range [0,1], and $W_i^t, W_i^{t-1}, W_i^{t-2}, W_i^{t-3}$ are the weight values at iterations t, $t-1$, $t-2$, and $t-3$.

Again, the weights W_{t+1}^{FB} are applied to the neural network function for attaining the neural network output, which is then used to compute the error.

$$Y^{FB} = F(r, W^{FB}) \tag{31}$$

$$E^{FB} = \frac{1}{N}\sum_{i=1}^{N}\left(Y_i^{FB} - Y_{Gi}\right) \tag{32}$$

Two errors E^{LM} and E^{FB} are calculated and the error having the least value is considered as the error value (E_{t+1}) at the current iteration, and its corresponding weight forms the final weights (W_{t+1}).

The errors computed at the current iteration E_{t+1} and the previous iteration E_t are compared and if the value is decreased, then μ is decreased with a factor v. In the other case, μ is increased by a factor v. This process is performed repetitively for T number of iterations, and the final weights are considered as the trained weights, which are utilized for speaker recognition for the received speech signal. Table 1 depicts the pseudo code of the FBLM algorithm.

Table 1. FBLM algorithm.

	FBLM Algorithm
1	**Input:** R → Feature matrix, Y_G → Ground truth
2	**Output:** w → Final weight vector
3	**Procedure**
4	**Begin**
5	Initialize the weight vector W_0
6	Determine the error E after the application of W_0 to $F(x, W)$
7	**While** $(t < T)$
8	Determine the Jacobian J
9	Determine H and g
10	Calculate new weight vector using equation (24)
11	Calculate the error E^{LM} after the application of W_{t+1}^{LM} to $F(x, W)$
12	Determine the new weight vector W_{t+1}^{FB} using equation (30)
13	Calculate the error E^{FB} after the application of W_{t+1}^{FB} to $F(x, W)$
14	If $(E^{LM} < E^{FB})$
15	$W_{t+1} = W_{t+1}^{LM}$ and $E_{t+1} = E^{LM}$
16	Else
17	$W_{t+1} = W_{t+1}^{FB}$ and $E_{t+1} = E^{FB}$
18	Endif
19	If $(E_{t+1} < E_t)$
20	Decrease μ with a factor v

```
            Else
                Increase $\lambda$ with a factor $v$
            Endif
        End while
        Return weight $W_{t+1}$
End
```

5. RESULTS AND DISCUSSION

Here, the experimental results of the proposed LFBNN classifier and the performance evaluation by comparing its performance with that of the existing methods using different evaluation metrics, like accuracy, False Rejection Rate (FRR), and False Acceptance rate (FAR), are presented.

5.1 Dataset description

The database taken for experimental analysis is English Language Speech Database for Speaker Recognition (ELSDSR) (Database; Feng, 2004). ELSDSR corpus of the speech is deliberated for providing speech data to creating and evaluating the automatic speaker recognition system. The faculty, master students, and Ph. D. students of the Department of Informatics and Mathematical Modelling (IMM) at Technical University of Denmark (DTU) jointly designed the ELSDSR corpus. The text language is in English, and it is read by one Canadian, one Icelander, and 20 Danes. Since a formal rehearsal has not been done, a perfect pronunciation is not obtained. However, it is not important to get the specific and unique characteristics from individuals. Here, the voice messages are recorded in '.wav' file type. PCM algorithm is used, and the sampling frequency is selected as 16 kHz with a bit rate of 16.

ELSDSR contains voice messages from 22 speakers (12 males +10 females). The speakers have the age range from 24 to 63. Most of the speakers are faculty and Ph. D. students working at IMM, and 5 of them are master students including one international master student. As the gender distribution is irregular at the experiment site, the average age of female speakers is higher than the average age of male speakers, and around half of the female subjects are secretaries in IMM. 84% of male speakers were between the age of 26 and 37 years, yet the ages of female speakers spread in a huge scale.

5.2 Evaluation metrics

The evaluation metrics considered for analyzing the performance of the proposed algorithm are accuracy, FRR, and FAR. The following equations define the performance metrics:

$$FAR = \frac{FP}{FP+TN} \tag{33}$$

$$FRR = \frac{FN}{FN+TP} \tag{34}$$

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FP} \tag{35}$$

where TP represents the True positive, FP represents the False positive, TN represents the True negative, and FN represents the False negative.

5.3 Experimental setup

The implementation of the proposed LFBNN algorithm is done using MATLAB (R2014a). The system has an i5 processor of 2.2GHz CPU clock speed with 4 GB RAM and 64-bit operating system running with Windows 8.1.

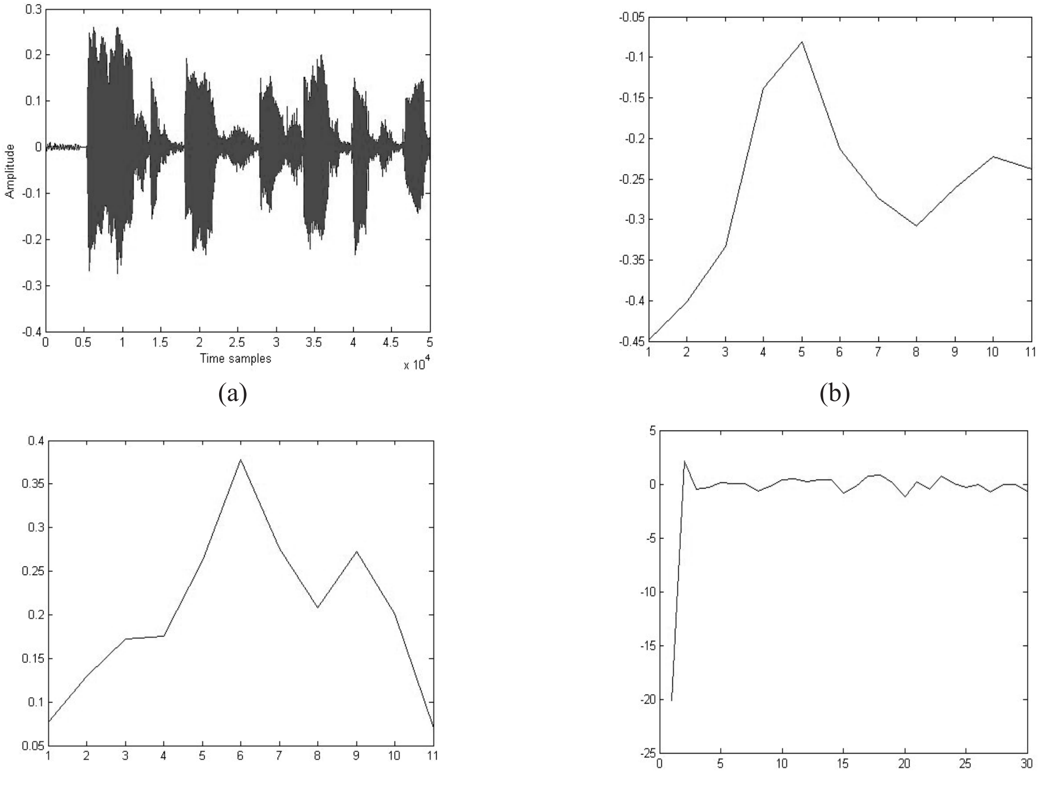
5.4 Parameters to be fixed

Parameters	Description
C	number of MFCC coefficients
p	number of LP coefficients
G	slope adjustment factor
E	number of hidden neurons

The performance of these parameters is evaluated with a variety of values, and the best value is selected for the comparison. The robustness of the proposed and existing algorithms is evaluated by injecting Additive White Gaussian Noise (AWGN) for the SNR of 0.1.

5.5 Experimental results

This section shows the experimental results of a speech signal with its feature extracted results. Figure 4.a depicts the speech signal for each time sample from 0 to 5x10^4. The plot obtained while extracting autocorrelation coefficients is sketched out in figure 4.b. The plots of predictivity ratio and MFCC coefficients for the speech signal are given in figure 4.a and are pictured out in figures 4.c and 4.d, respectively. For the speech data given as input for speaker recognition, a spectral centroid of 1905.4 is obtained. Finally, it can be shown that the proposed LFBNN method is effective in performing speaker recognition.



(a)

(b)

(c)

(d)

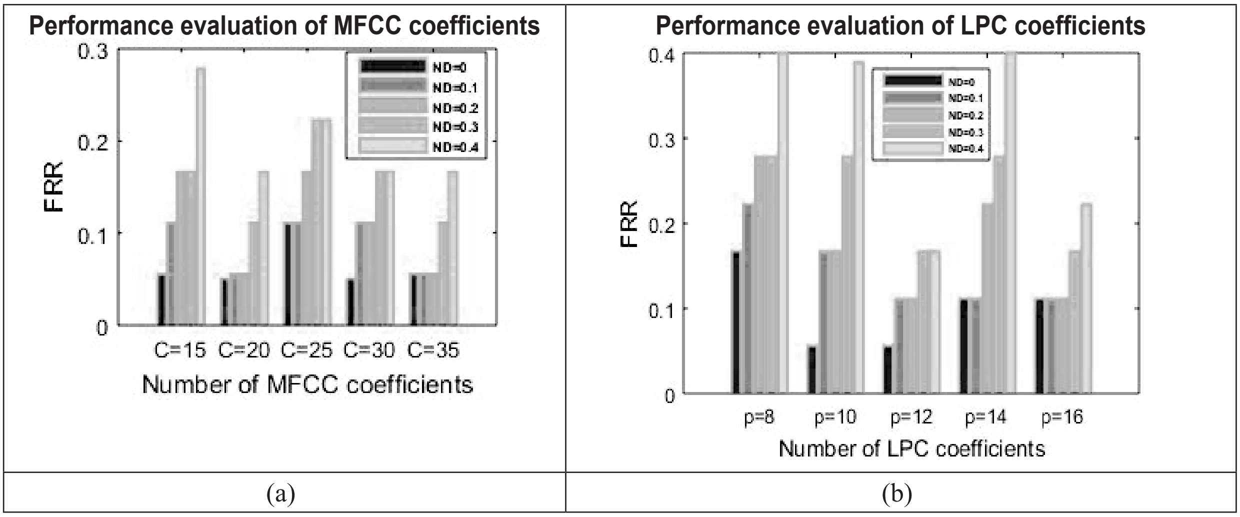
Figure 4. Experimental Results: (a) Speech signal; (b) Plot of autocorrelation coefficients; (c) Plot of predictivity ratio; and (d) Plot of MFCC coefficients.

5.6 Performance Analysis

This section shows the performance analysis of the proposed algorithm to define the optimal parameters for the applicability.

a) Analysis based on FRR

Figure 5 shows the performance graph based on FRR for the different number of MFCC coefficients; LPC coefficients for the noise density ND varied as 0, 0.1, 0.2, 0.3, and 0.4. Figure 5.a shows the FRR analysis graph for varying MFCC coefficients, where the minimum value of FRR found is 0.05 for ND= 0 with MFCC coefficients, denoted as C, as 20 and 30, respectively. At maximum ND, a minimum FRR of 0.1667 is obtained for C=20, 30, and 35. In figure 5.b, the analysis result for p varying from 8, 10, 12, 14, and 16 is presented, for ND ranging from 0 to 0.4. When ND=0.4, the least FRR obtained is 0.1667 for p=12.


Performance evaluation of MFCC coefficients

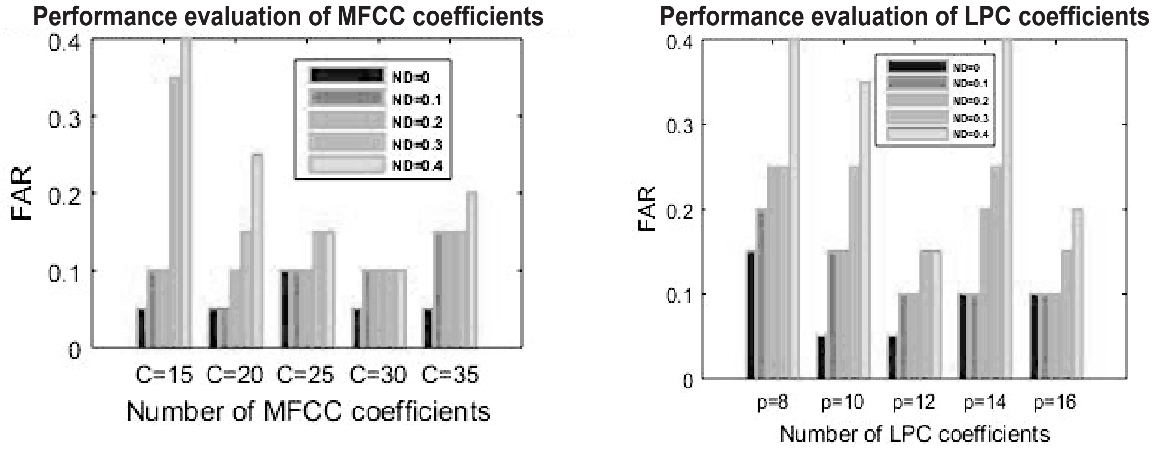
Performance evaluation of LPC coefficients

(a) (b)

Figure 5. FRR Analysis: a) based on MFCC coefficients; b) based on LPC coefficients.

b) Analysis based on FAR

In Figure 6, the analysis based on FAR is demonstrated for varying MFCC coefficients, LPC coefficients for different noise densities. As shown in figure 6, it is seen that FAR is minimum for ND=0 for all the four parameters considered. Figure 6.a shows the FAR analysis graph for C=15, 20, 25, 30, and 35 with ND ranging from 0 to 0.4. When ND=0.1, the FAR obtained is 0.1, 0.05, 0.1, 0.1, and 0.15 for C= 15, 20, 25, 30, and 35, which increases when ND=0.4. Figure 6.b pictures out the FAR analysis chart for different values of p, where the least FAR value found is 0.15 at the maximum value of ND at p=12.

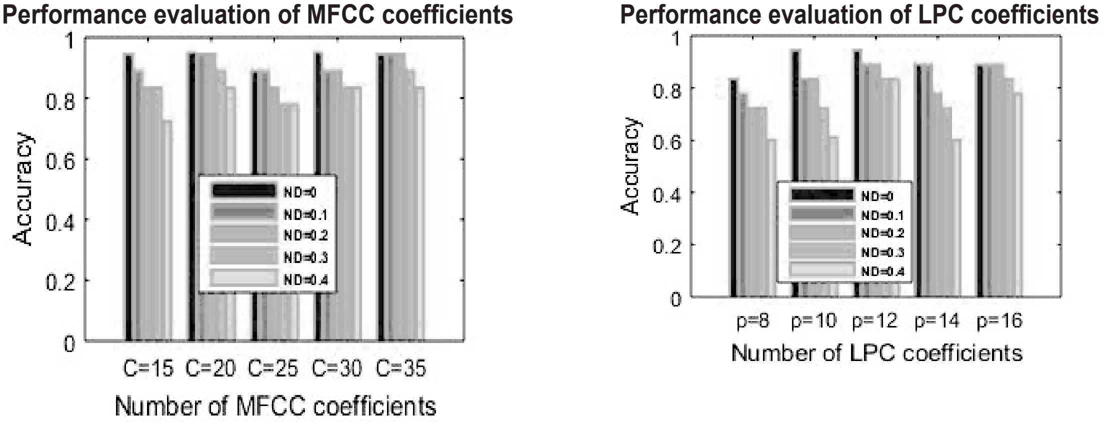


(a) (b)

Figure 6. FAR Analysis: a) based on MFCC coefficients; b) based on LPC coefficients.

c) Analysis based on Accuracy

The performance results of analysis made based on accuracy are shown in figure 7 for different MFCC coefficients, LPC coefficients. At minimum noise density, the accuracy is found to be in maximum, for all the parameters. In figure 7.a, the accuracy analysis chart for different C values is depicted by varying the ND values. The maximum accuracy achieved in this case is 95% when ND=0 at C= 20, 30. As ND is fixed as 0.4, the accuracy is reduced to 83.33% for the same values of C. Figure 7.b presents the analysis result based on accuracy for different p values, where the maximum accuracy obtained is 94.44% for p=10 and p=12 when the noise density is 0.



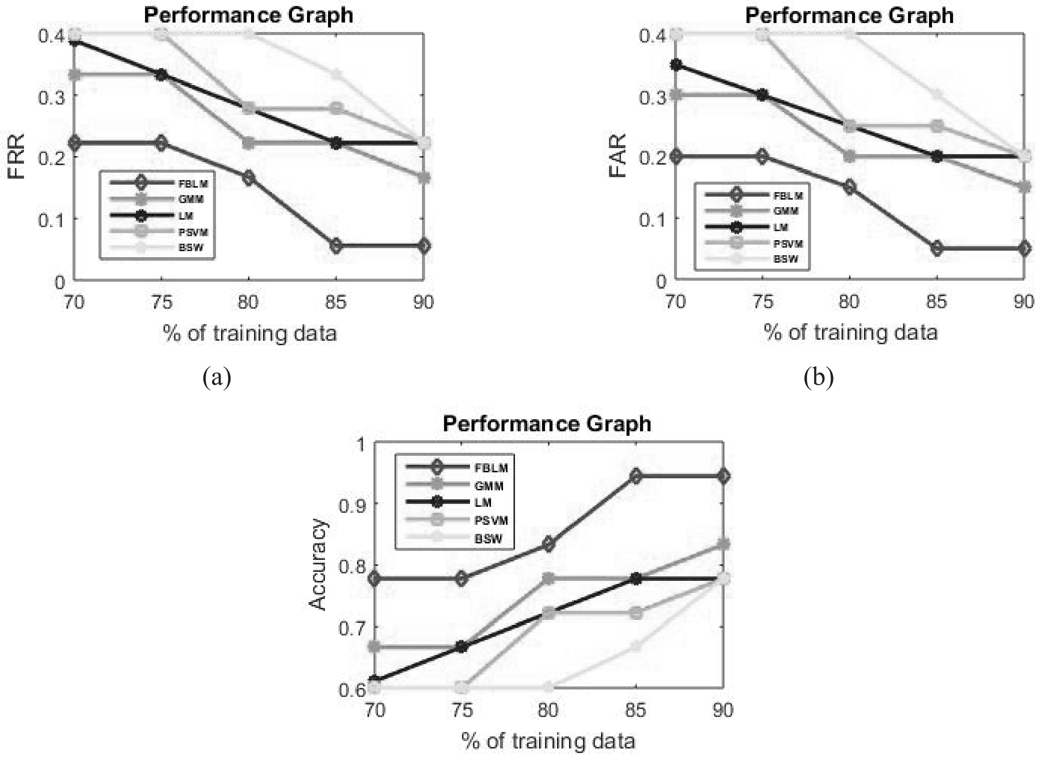
(a) (b)

Figure 7. Accuracy Analysis: a) based on MFCC coefficients; b) based on LPC coefficients.

5.7 Comparative analysis

The best parameter values determined from the performance evaluation are fixed to the proposed FBLM algorithm, and the analysis is performed on various percentages of training data to obtain FAR, FRR, and accuracy curve. In the original signal analysis, the direct signal is given as input to the feature extraction method, and the recognition is carried out using the proposed method. The existing methods employed for the comparison are GMM (Clanahan & Leon, 2015), LM (Moeikham & Srinonchat, 2008), PSVM (Cumani &Laface, 2014), and BSW (Sadjadi & Hansen, 2014).

In figure 8, the result of the comparative analysis carried out without noise is shown for the percentage of data varying from 70 to 90. In figure 8.a, the FRR analysis graph of all the comparative techniques is shown. The FRR value obtained in the proposed FBLM is 0.222 initially, which reduces to 0.056, as the percentage of data is raised. Meanwhile, BSW has 0.222 as the minimum FRR. Figure 8.b presents the result of analysis based on FAR in FBLM, GMM, LM, PSVM, and BSW, for varying percentages of data. When the existing GMM approach has FAR of 0.15, the proposed FBLM method has only 0.05 as the FAR value. The comparative analysis chart based on accuracy is pictured out in figure 8.c. As shown in figure 8.c, when the training percentage of data increases, the accuracy also increases. Initially, FBLM has an accuracy of just 77.778%, which increases to 94.44% as the data percentage is 90. Meanwhile, the maximum accuracy achieved among the existing techniques is 83.33%, by GMM.



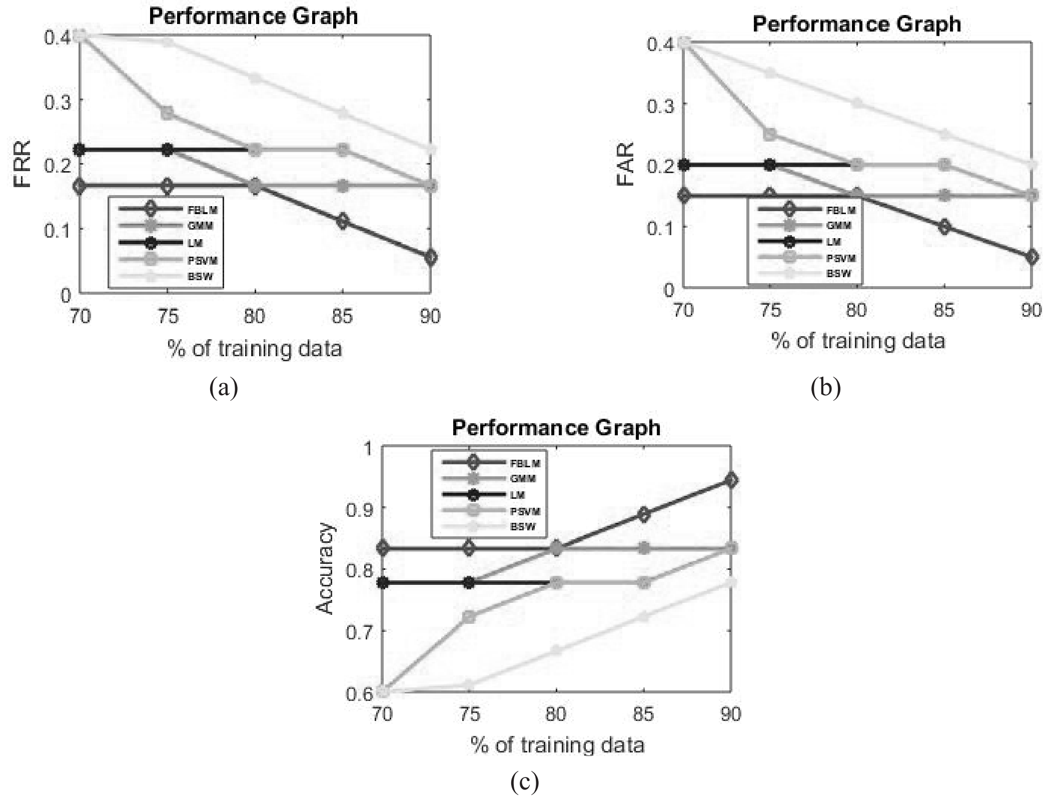
(a)

(b)

(c)

Figure 8. Analysis without noise: a) FRR; b) FAR; c) Accuracy.

The result of analysis after transmitting of the signal through AWGN channel is presented in figure 9, based on the three evaluation metrics. Figure 9.a shows the FRR graph obtained in the comparative techniques by varying the percentages of training data from 70 to 90. With 70% training data, the FRR obtained in GMM and LM is 0.222, and that in PSVM and BSW is 0.4, whereas in FBLM, it is 0.1667. The minimum FRR value attained by the proposed FBLM method is 0.056, while in the exiting GMM, LM, and PSVM, it is 0.1667, respectively. In figure 9.b, the graph of FAR with noisy information is depicted. For 90% of the training data, the proposed FBLM achieved the minimum FAR of 0.05. Meanwhile, GMM, LM, and PSVM have FAR of 0.15. The accuracy graph with noise content added is given in figure 9.c. When the proposed FBLM method has an accuracy of 94.44% at a maximum percentage of training data, the existing GMM, LM, and PSVM have only 83.33% as their maximum accuracy. In this case, the accuracy attained by BSW is just 77.778%. The comparative results clearly show that the proposed algorithm is more robust than the existing techniques considered.



(a)

(b)

(c)

Figure 9. Analysis with noise: a) FRR; b) FAR; c) Accuracy.

5.8 Discussion

A discussion is made regarding the accuracy rate attained in the comparative techniques together with few more existing techniques, like Discrete Cosine Transform (DCT) (Kekre *et al.,* 2010), Walsh (Kekre *et al.,* 2010), Haar (Kekre *et al.,* 2010), Vector Quantization (VQ) (Al-Shboul *et al.,* 2007), Multi-Layer Perceptron (MLP) (Subha & Kannan, 2015), Linear Discriminant Analysis (LDA) (Subha & Kannan, 2015), Hidden Markov Model (HMM) (Subha & Kannan, 2015), Kaiser (Jain & Sharma, 2014), and Rectangular (Jain & Sharma, 2014), as listed in table 2. As shown in the table, the rate of accuracy attained by the proposed FBLM is 95%, whereas the maximum accuracy that could be attained by the existing method is 94.444% by GMM and LM. Meanwhile, the windowing techniques, Kaiser and Rectangular, have an accuracy of just 86.363% and 84.09%, respectively. The techniques LDA and MLP have an accuracy of 70.7% and 82.1%, which is lower than that of the windowing techniques. Thus, from the discussion table, it is seen that the proposed FBLM algorithm outperforms the existing techniques compared with 95% accuracy.

Table 2. Performance comparison based on accuracy analysis.

Methods	*Accuracy (%)*	*Methods*	*Accuracy (%)*
FBLM	95	***Haar***	86.39
LM	94.44	***VQ***	89.1
PSVM	88.88	***LDA***	70.5
BSW	83.33	***MLP***	82.1
GMM	94.44	***HMM***	85
DCT	88.05	***Kaiser***	86.36
Walsh	85.55	***Rectangular***	84.09

6. CONCLUSION

In this paper, a new classifier model, called LFBNN, is developed, for the robust and hybrid training of FFNN with hybrid features-enabled speaker recognition system. In feature extraction, four different features, that is, autocorrelation coefficients and predictivity ratio MFCC and spectral centroid, were utilized. In the classification step, the neural network was trained based on the proposed training algorithm FBLM. The proposed training algorithm is developed by integrating the LMA and Fractional Bat Algorithm. For the experimentation, ELSDSR database is taken, and the performance evaluation was done using different evaluation metrics, like FAR, FRR, and accuracy. In the performance evaluation, the detailed analysis is carried to find out the better parametric value for a different number of MFCC coefficients, LP coefficients, slope adjustment factor, and hidden neurons. The comparative analysis is also performed for the proposed algorithm with the GMM model to prove the recognition accuracy with and without robustness. From the experimental analysis, it is seen that the proposed method obtains an accuracy of 95% and thus proved that the proposed algorithm is vigorous than the existing techniques. In the future, the new development of features and optimal estimation of the classifier model can provide improved performance.

REFERENCES

Alku, P & Saeidi, R. (2017). The Linear Predictive Modeling of Speech From Higher-Lag Autocorrelation Coefficients Applied to Noise-Robust Speaker Recognition,' *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **25**(8): 1606-1617.

Al-Shboul, B., Alsawalqah, H. & Lee, D. (2007). Real-time speaker identification system', *in Proceedings of the 7th WSEAS International Conference on Applied Computer Science*, November, 424-428.

Avci D. (2009). An expert system for speaker identification using adaptive wavelet sure entropy', *Expert Syst. Appl*, **36**: 6295–6300.

Burred, J.J. & Lerch, A. (2003). A hierarchical approach to automatic musical genre classification', *in proceedings of 6th International Conference on Digital Audio Effects (DAFx-03)*, London, UK, 8-11.

Campbell, J., Shen, W., Campbell, W., Schwartz, R., Bonastre, J.F & Matrouf, D. (2009). Forensic speaker recognition: A need for caution,' *IEEE Signal Processing Mag*, **26**(2): 95-103.

Campbell, W., Campbell, J., Reynolds, D., Singer, E & Torres-Carrasquillo, P.A. (2006). Support vector machines for speaker and language recognition' *Comput. Speech Lang*, **20**(2-3): 210-229.

Clanahan, R.M. & Leon, P.L.D. (2015). Reducing computation in an i-vector speaker recognition system using a tree-structured universal background model', *Speech Communication*, **66**: 36-46.

Cumani, S & Laface, P. (2014). Large-Scale Training of Pairwise Support Vector Machines for Speaker Recognition', *IEEE/ACM transactions on audio, speech, and language processing*, **22**(11): 1590-1600.

Cumani, S & Laface, P. (2018). Speaker Recognition Using e–Vectors,' *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **26**(4): 736-748.

El-Melegy, M.T. (2013). Random Sampler M-Estimator Algorithm With Sequential Probability Ratio Test for Robust Function Approximation Via Feed-Forward Neural Networks', *Neural Networks and Learning Systems, IEEE Transactions* on, **24**(7):1074-1085.

English Language Speech Database for Speaker Recognition (ELSDSR) from "http://www2.imm.dtu.dk/~lfen/elsdsr/@

Feng, L. (2004). Speaker Recognition, Informatics and Mathematical Modelling', *Technical University of Denmark*, DTU.

Fischer, J. (2013). Adaptive reduction of noise signals and background signals in a speech-processing system.*' U.S. Patent, 8, 352,256,* 8.

Ghahabi, O & Hernando, J. (2017). Deep Learning Backend for Single and Multisession i-Vector Speaker Recognition,' *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **25**(4): 807-817.

Jain, A & Sharma, O.P. (2014). Evaluation of MFCC for speaker verification on various windows', *in proceedings of International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*, 1-6.

Jain, A ., Ross, A & Prabhakar, S. (2004). An introduction to biometric recognition,' *IEEE Trans. Circuits Systems Video Technol*, **14**(1): 4-20.

Kekre, H.B., Sarode, T.K., Natu, S.J. & Natu, P.J. (2010). Speaker Identification Using 2-D DCT, Walsh and Haar on Full and Block Spectrogram', *International Journal on Computer Science and Engineering*, **02**(05): 1733-1740.

Kinnunen, T & Li, H. (2010). An overview of text-independent speaker recognition: From features to supervectors,' *Speech Commun*, **52**(1): 12-40.

Kua, J.M.K., Thiruvaran, T., Nosratighods, M., Ambikairajah, E. & Epps, J. (2010). Investigation of Spectral Centroid Magnitude and Frequency for Speaker Recognition', *The Speaker and Language Recognition Workshop*.

May, T., Par, S.V.D & Kohlrausch, A. (2012). Noise-Robust Speaker Recognition Combining Missing Data Techniques and Universal Background Modeling', *IEEE transactions on audio, speech, and language processing*, **20**(1): 108-121.

Ming, J., Hazen, T.J., Glass, J.R & Reynolds, D.A. (2007). Robust speaker recognition in noisy conditions,' *IEEE Trans. Audio Speech Language Process*, **15**(5): 1711-1723.

Moeikham, P & Srinonchat, J. (2008). Adjusted Levenberg - Marquardt technique for improvement speech recognition system, *in proceedings of 9th International Conference on Signal Processing*, Beijing, 575-578.

Morales-Cordovilla, J.A., Peinado, A.M., Sánchez, V. & González J.A. (2011). Feature Extraction Based on Pitch-Synchronous Averaging for Robust Speech Recognition', *IEEE transactions on audio, speech, and language processing*, **19**(3): 640-651.

Nath, D. & Kalita, S.K. (2014). Feature Selection Method for Speaker Recognition using Neural Network', *International Journal of Computer Applications,* **101**(3): 0975-8887.

On, C.K., Pandiyan, P.M., Yaacob, S. & Saudi, A. (2006). Mel-Frequency Cepstral Coefficient Analysis in Speech Recognition', *in proceedings of International Conference on Computing & Informatics*, 1-5.

Pan, F., Jie, C., Xuyan, T. & Jiwei , F. (2005). Multilayered feed forward neural network based on particle swarm optimizer algorithm', *Systems Engineering and Electronics, Journal of*, **16**(3): 682-686.

Pujol, J. (2007). The solution of nonlinear inverse problems and the Levenberg-Marquardt method', *eophys. J. Int*, 72(4).

Pullella, D., Kuhne, M & Togneri, R. (2009). Sub-band partitioning for full covariance based missing data speaker recognition,' *Int. J. Inform.Syst. Sci., (Advances in Information and Systems Sciences Series),* **3**(3-4): 641-648.

Ranjan, S & Hansen, J.H.L. (2018). Curriculum Learning Based Approaches for Noise Robust Speaker Recognition,' *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **26**(1): 197-210.

Reynolds, D.A., Quatieri, T.F. & Dunn, R.B. (2000). Speaker verification using adapted Gaussian mixture models', *Digit. Signal Process*, **10**(1-3): 19-41.

Sadjadi, S.O & Hansen, J.H.L. (2012). Blind Reverberation Mitigation for Robust Speaker Identification', *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4225-4228.

Sadjadi, S.O & Hansen, J.H.L. (2014). Blind Spectral Weighting for Robust Speaker Identification under Reverberation Mismatch', *IEEE/ACM transactions on audio, speech, and language processing*, **22**(5): 937-945.

Sahidullah, M & Saha, G. (2013). A Novel Windowing Technique for Efficient Computation of MFCC for Speaker Recognition', *IEEE signal processing letters*, **20**(2): 149-152.

Subha, S & Kannan, P. (2015). Speaker Identification Techniques–A Survey', **4**(10):190-193.

Thomas R & Rangachar, M.J.S. (2017). Fractional Bat and Multi-Kernel-Based Spherical SVM for Low Resolution Face Recognition,' *International Journal of Pattern Recognition and Artificial Intelligence*, **31**(8):1-28.

Togneri, R & Pullella, D. (2011). An Overview of Speaker Identification: Accuracy and Robustness Issues',*Circuits and Systems Magazine,* IEEE, Biometrics Compendium, **11**(2): 23-61.

Wu, J.D. & Lin, B.F. (2009). Speaker identification using discrete wavelet packet transform technique with irregular decomposition, *Expert Syst. Appl*, **36**: 3136-3143.

Xu, L., Lee, K.A., Li, H & Yang, Z. (2018). Generalizing I-Vector Estimation for Rapid Speaker Recognition,' *IEEE/ACM Transactions on Audio, Speech, and Language Processing,* **26**(4): 749-759.

(Saquib, et al., 2010) Saquib, Z., Salam, N., Nair, R.P., Pandey, N & Joshi. A., (2010). A Survey on Automatic Speaker Recognition Systems, *Communications in Computer and Information Science, 123. Springer, Berlin, Heidelberg.*

(Variani, et al., 2014) Variani, E., Lei, X., Dermott, E.M. & Moreno, I.L (2014). Gonzalez-Dominguez, "Deep Neural Networks for Small Footprint Text-Dependent Speaker Verification, In Proceedings of IEEE International Conference on Acoustics, *Speech and Signal Processing*, 4052-4056.

(Torfi, et al., 2017) Torfi, A., Dawson, J & Nasrabadi. N.M., (2017). Text-Independent Speaker Verification using 3d Convolutional Neural Networks, *Computer Vision and Pattern Recognition.*

(Song, et al., 2018) Song, S., Zhang, S., Schuller, B.W., Shen, L & Valstar. M., (2018). Noise Invariant Frame Selection: A Simple Method to Address the Background Noise Problem for Text-independent Speaker Verification, *Computer Vision and Pattern Recognition.*

(Zhang, et al., 2017) Zhang, C., Yu, C & Hansen. J.H.L., (2017). An Investigation of Deep-Learning Frameworks for Speaker Verification Antispoofing, *IEEE Journal of Selected Topics in Signal Processing,* **11**(4): 684-694.