Speaker Recognition Based on Mutated Monarch Butterfly Optimization Configured Artificial Neural Network

Original Scientific Paper

Dhana Lakshmi Namburi
Research Scholar, ANU, Guntur, Andhra Pradesh, India
Assistant Professor, CBIT, Hyderabad, Telangana, India
dhananm@gmail.com

Satya Sai Ram M
RVR & JC College of Engineering,
Chowdavaram, Andhra Pradesh, India
Msatyasairam1981@gmail.com

Abstract *– Speaker recognition is the process of extracting speaker-specific details from voice waves to validate the features asserted by system users; in other words, it allows voice-controlled access to a range of services. The research initiates with extraction features from voice signals and employing those features in Artificial Neural Network (ANN) for speaker recognition. Increasing the number of hidden layers and their associated neurons reduces the training error and increases the computational process's complexity. It is essential to have an optimal number of hidden layers and their corresponding, but attaining those optimal configurations through a manual or trial and the process takes time and makes the process more complex. This urges incorporating optimization approaches for finding optimal hidden layers and their corresponding neurons. The technique involve in configuring the ANN is Mutated Monarch Butterfly Optimization (MMBO). The proposed MMBO employed for configuring the ANN achieves the sensitivity of 97.5% in a real-time database that is superior to contest techniques.*

Keywords*: Speaker recognition, Speaker verification, Speaker identification, Artificial Neural Network, Monarch Butterfly Optimization, Model configuration.*

1. INTRODUCTION

Since a decade ago, academics and industry have paid increasing attention to speaker identification [1]. It is extensively used in applications, including security and surveillance, financial security, discriminative speaker embedding learning, voice authentication, forensic voice verification for suspect detection [2], electronic voice eavesdropping, voice conversion, and identity verification, as well as access control, biometrics authentication, mobile shopping, and mobile banking [3]. It essentially involves classifying unknown speakers based on their speech [4]. Speaker identification is the process of identifying a speaker sound based on a set of trained speaker sounds. In other words, speaker identification compares one user's voice profile with many other profiles and determines the best or exact match. Since speech signals are the primary means of communication, they constantly contain rich, relevant details, such as speakers' accents, gender, emotions, and other characteristics. As a result of these distinctive characteristics, researchers can distinguish between speakers during phone calls, even when the speakers are not physically present [6] [8] [9].

Speaker Identification involves identifying unknown voices from a fixed set of known speakers. Therefore, it is called closed set identification. Based on the speech used for identifying the speaker, the systems can be grouped into text-dependent (fixed text is used for both training and testing phase) and text-independent (no fixed text). Out of the two types, text-independent specker recognition is most challenging job. The error that can occur in speaker identification is false identification, which can be measured by sensitivity, which determines the correctness of the predictions. A high sensitivity model provides a more reliable result than a low sensitivity model in medical applications. Hence, the objective of this work is to build a model for text independent speaker recognition with improved recognition accuracy as well as sensitivity.

A variety of models, techniques, and algorithms are employed to identify speakers in recent literature, including Mel-frequency Cepstral Coefficients (MFCC) and Linear Predictive Coding (LPC) [10], the Histogram Transform Model [11], and spatiotemporal sparse coding and hierarchical pooling [12]. In HT based SI systems achieves identification accuracy of 99.52% and is affected by H (random affine transformations). Increasing H improves the identification accuracy, but when H is higher than 400, the accuracy decreases instead. Similarly in Visual speaker identification and authentication by joint spatiotemporal sparse coding and hierarchical pooling archives higher identification accuracy with the increase of dictionary size K. As K increases results in very high computational complexity and large memory cost during the classifier training process.

Despite their effectiveness and accuracy, these traditional speaker identification methods have not been able to identify human voices effectively. A method based on Artificial Intelligence (AI) has been proposed by speech processing researchers to overcome this issue [13]. In recent years, AI technology has substantially enhanced both the recognition rate and robustness of speaker identification. As a result, the results produced by machine learning neural networks continue to support neural networks' use in speaker identification. Support vector machines (SVMs), artificial neural networks (ANNs), and K-nearest neighbours (KNNs) are among the methods used to identify speakers in literature. Among these, ANNs have proven effective in identifying speakers. Over the last three decades, ANNs have also been extensively studied and applied to classification, pattern recognition, regression, and forecasting.

Despite its numerous advantages, traditional ANNs still lack accuracy and performance. Therefore, placing the optimal number of hidden layers enhances traditional ANN performance. In recent years, MBO (Monarch Butterfly Optimization) procedures have been proposed in various literature [18] [19], so the research employs MBO. The search strategy of the basic MBO algorithm, on the other hand, readily slips into local optima, resulting in precocious convergence and low performance on many complicated optimization tasks. Scholars have made several enhancements to MBO in recent years to improve its effectiveness [20][21]. However, these techniques do have not a sufficient performance in view of convergence speed and accurate optimum solution. To solve the issues, this paper develops an Oppositional based strategy with the Cauchy distribution (Cd) technique in MBO is proposed. First, is the Opposition Based Learning model, which ensures the exploration of unique and opposing candidate solutions in the search space while the evolution process is ongoing in order to assess the better candidate solutions [22] [23]. Secondly, Cd as a mutation operator enriches the conventional performance MBO algorithm [24].

2. LITERATURE REVIEW

Daqrouq et al. (2015) [25] had proposed a speaker recognition system that utilizes a combination of formants, wavelets Entropy, and neural network classifiers to identify vowels characteristics. The initial stage involved extracting five formants and seven Shannon entropy wavelet packets from the speakers' signals to build the speaker feature vector. In the next stage, these 12 feature extraction coefficients were utilized as inputs to feed-forward neural networks. The suggested technique performs well in speaker verification and identification tasks, according to the findings of the experiments. The results were shown to be superior to well-known classical speaker detection techniques.

Faragallah, Osama S et al. (2018) [26] had proposed MKMFCC–SVM is a robust noisy automated speaker identification (ASI) technique. It uses a support vector machine and the Multiple Kernel Weighted-MFCC (MKMFCC). In the face of noise or deterioration, experimental studies showed that the suggested MKMFCC–SVM ASI method gives a greater identification rate.

Chen et al. (2019) [27] had proposed a bi-level framework to mutually optimize session compensation and support vector machine (SVM) based classifier for speaker identification. Finally, the trials demonstrated that in the i-vector framework, the proposed techniques outperformed existing session compensation algorithms and classifiers.

de Abreu Campos et al. (2019) [28] had proposed an unsupervised learning technique such as RL-Sim and ReckNN for speaker retrieval and recognition. The method was organised around a framework that makes use of a rank-based formulation. The adoption of unsupervised learning algorithms over standard speaker identification approaches resulted in effectiveness enhancements of up to +56 percent on retrieval measures.

Safavi et al. (2018) [29] had proposed Automatic identification of the speaker, age group, and gender from children's speech. A number of classification techniques were examined, including the Gaussian Mixture Model–Universal Background Model, GMM–SVM, and i-vector established systems. As one might imagine, the mistake rate for speaker recognition lowers with age. However, the influence of age on gender and age-group documentation was more complicated, owing to the repercussions of adolescent. Finally, the ability of distinct bandwidths to identify speakers, age groups, and gender from children's speech was tested.

Devi et al., (2020) [30] had proposed a hybrid technique for Automatic Speaker Recognition that uses speech signals and an ANN to increase speaker prediction accuracy. The proposed ANN-based approach was designed based on Multilayer Perceptron (MLP) with Bayesian Regularization. In contrast to existing models, the suggested strategy was validated by performance assessment and classification accuracies. The authors claimed that the suggested method provided a nicer recognition rate and 93.33% accuracy was achieved.

Biswas et al. (2021) [31] had proposed a multi-layer perceptron neural network to identify singers' voices. The trials for singer identifications were repeated five times in this study, and the analysis was carried out using feature extraction. Apart from the employment of the supervised learning approach with the insinuation of weight optimization, the effectiveness was found for the recognition of the novel and unidentified vocalist to be discovered. Finally, the study found that the identification was accurate to the tune of 99.29%.

Wang et al. (2019) [32] had proposed a MBO is a nature-inspired metaheuristic algorithm inspired by monarch butterfly migratory behaviour. As a result, the monarch butterfly's locations were updated in two ways. The offspring were first created (position updating) by the migration operator, which may be changed by the migration ratio. The butterfly regulating operator is then utilized to fine-tune the locations of other butterflies. In comparison to previous algorithms, the MBO technique convincingly demonstrated its capacity to discover increased function values on majority of the benchmark issues.

Chakraborty et al. (2019) [33] had proposed by using Oppositional Based Learning (OBL) and Dynamic Cauchy Mutation (DCM), an Enhanced Elephant Herding Optimization (EEHO) to address the multilevel image thresholding issue for image segmentation. OBL improves the performance of normal EHO by speeding up the convergence rate, whereas DCM prevents premature convergence. This proposed algorithm delivers capable performance equated to other methods.

3. PROPOSED METHODOLOGY

Speaker recognition is a method for recognizing who is speaking automatically by utilizing speaker-specific information included in voice waves. The voice signal contains critical information such as message content, language, speaker identification, emotion, personality, and so on. It permits voice-controlled access to various services.

To build a Speaker Identification (SI) System, the model parameters are regularly learned based on the features extracted from the speech samples in the training phase. Testing involves feeding the extracted features from unknown speech to the trained model to identify who is speaking. Most widely used feature extraction methods are MFCC and LPC. The MFCC is a popular feature in Automatic Speech Recognition (ASR) and is inferred following static (non-signal dependent) processing methods. A LPC gives a decent model of the speech signal. This is particularly valid for the quasi-steady state voiced regions of speech in which all-pole model of LPC give a good approximation to the vocal tract spectral envelope. Different classifiers are likewise accessible for SI namely Kernel Regression and K Nearest Neighbour (KNN), Support Vector Machine (SVM), Hidden Markov Model (HMM), Maximum Likelihood Classifier (MLC) and ANN.

This research initiates with extraction features from voice signals and employing those features to Artificial Neural Network (ANN) for speaker recognition. It is also possible to reduce the training error by increasing the number of hidden layers and their linked neurons, as well as increasing the complexity of the computational process. In ANN, the hidden layer is critical for identifying characteristics in the input data and using them to correlate between a given input and the proper output. A higher number of hidden layers increases the order of weights, and it helps to make a higher-order decision boundary. Similarly, increasing hidden layers would also increase the complexity of the model and sometimes lead to over-fitting. It is essential to have an optimal number of hidden layers and their corresponding, but attaining those optimal configurations through a manual or trial and process takes time and makes the process more complex. These urges incorporate optimization methods for recognizing optimal hidden layers and their corresponding neurons. The technique involves in configuring the ANN is MMBO; the process of integrating the opposition strategy and the Cd strategy to enhance the performance of traditional MBO. The study used 200 real-time speech signal datasets from 20 speakers, including eight female and twelve male voice signals for 10 words each. ANN operates with Levenberg–Marquardt (LM) as a training technique for speaker recognition. This research considers 80% of datasets for training and the remaining 20% for testing the configured model.

The flow diagram of the research work is shown in figure 1. It consists of two stages. Stage 1 is training the model; Stage 2 is testing the model. In training process, Features are extracted from training data set and these features are used to train the ANN. ANN architecture is modified by optimizing the number of hidden layers and number of hidden layer neurons by using different optimization techniques.

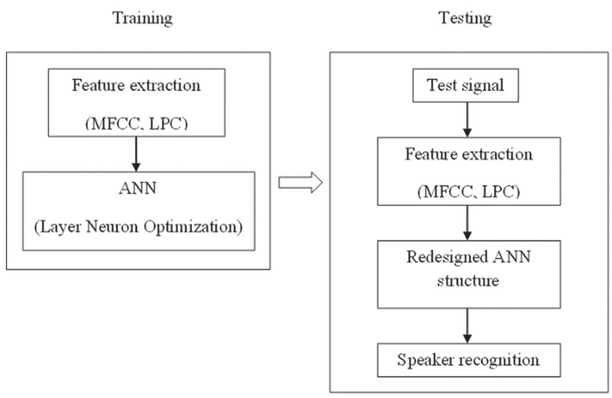

Fig. 1. Flow diagram of the process

3.1. FEATURE EXTRACTION

Feature extraction is important for speaker voice identification; this procedure is carried out using a couple of well-known algorithms named MFCC and LPC.

3.2. ARTIFICIAL NEURAL NETWORK (ANN)

ANN is adaptive and dynamic, learning and altering in response to each unique internal or exterior input. ANNs are employed in systems for sequence and pattern identification, data processing, robotics, and modeling.

3.2.1. MONARCH BUTTERFLY OPTIMIZATION (MBO)

Wang introduced the MBO algorithm in 2015, which is a type of swarm intelligence meta-heuristic procedure inspired by monarch butterfly migratory behavior. Individuals in MBO are updated through the migration and butterfly adjustment operations. When tackling global numerical optimization, the MBO outperforms numerous state-of-the-art optimization approaches. The migratory operator and the butterfly-adjusting operator are used to update the locations of monarch.

Initialization

The initialization of randomly generated solutions is the first step in every optimization approach. In the ranges of 1 to 5 and 1 to 30, the number of hidden layers and the values of their corresponding neurons are created at random. The length of the solution is determined by the value of the hidden layer created randomly in the first location. For example, if the value of the randomly allocated hidden layer is 3, the solution length will be 4 (3+1). Similarly, produce 10 solutions at random and feed them into the fitness process to assess the solution's strength. The primary purpose of opposition-based solution generation is to evaluate matching opposing estimates as a subsequent set of candidate solutions in order to improve the present candidate solution's approximation. An opposing candidate solution has been shown to have a higher likelihood of being nearer to the global optimal solution than a randomly picked candidate solution. Mathematically opposition based solution generation expressed as,

$$N^{o}_{(j,i)} = x_i + y_i - N_{(j,i)} \tag{1}$$

Let N ∈ [a, b] be a real numbers; Where, N^o is the opposition based solution and *N* refers randomly generated solution and x_i, y_i refers the minimum and the maximum values respectively. The both randomly generated solution and the opposition based solution generations are fed in to fitness computation for process evaluation.

Fitness Function

The fitness approach to evaluate how well a solution performs in comparison to the overall amount of validation data.

$$\text{Fitness} = \frac{\text{Correctly Recognize}}{\text{Total Number of Validation Data}} \tag{2}$$

Migration Operator

The monarch butterfly migration between Lands 1 and 2 is expected to be updated by the migration operator, with monarch butterflies solely belonging to subpopulations 1 and 2. Initially, NP1 = *ceil (p*NP) and NP2 = NP-NP1* may be used to compute the number of monarch butterflies in Lands 1 and 2.

Where NP represents the total number of monarch butterflies in Land 1, p denotes the monarch butterfly ratio in Land 1, and *ceil(y)* represents the rounding of y to the nearest whole number larger than or equal to y. In this way, migration operator arranged as

$$y^{t+1}{}_{i,k} = \begin{cases} y^{t}{}_{r1,k} & | r \le p \\ y^{t}{}_{r2,k} & | r > p \end{cases} \tag{3}$$

Wherever $y^{t+1}_{i,k}$ denotes the k^{th} element of y_i at generation t+1. Basically, $y^t_{r1,k}$ demonstrates the k^{th} element of y_{r1} at generation *t*, and $y^t_{r2,k}$. denotes the kth element of y_{r2} at generation *t*. *t* represents the current generation number. Monarch butterflies (r1 and r2) are arbitrarily selected from subpopulations 1 and 2. The condition variable (C_r) is found as follows:

$$C_r = rand * m_{tf} \tag{4}$$

The fundamental MBO technique is as follows: m_{tf} is the migration time frame, which is set to 1.2, and rand is a arbitrary number derivative from a uniform distribution.

Butterfly Adjusting Operator

The locations of monarch butterflies in subpopulation 2 are updated utilizing this operator. It may be updated as follows:

$$y^{t+1}{}_{j,k} = \begin{cases} y^{t}{}_{best,k} & | rand \le p \\ y^{t}{}_{r3,k} & | rand > p \end{cases} \tag{5}$$

Where $y^{t+1}_{j,k}$ denotes the k^{th} element of y_j at generation *t + 1;* $y^t_{best,k}$ denotes the kth element of ybest at generation *t*, this indicates the finest monarch butterfly habitat in Lands 1 and 2. The $y^t_{r3,k}$ denotes the kth element of y_{r3} at generation *t*; the monarch butterfly *r3* is randomly selected from subpopulation 2. If *rand > p*, there is a different development. If *rand > BAR*, the butterfly's position is also updated using Levy flying:

$$y^{t+1}{}_{i,k} = y^{t+1}{}_{j,k} + \alpha \times (dy - 0.5) \tag{6}$$

The variable BAR stands for the Butterfly Adjusting Rate; if BAR is less than a arbitrary value, the kth element of y_j at generation *t*+1 is changed, where is the weighting factor, as exposed in Equation (7).

$$\alpha = WS_{\max} / t^2 \tag{7}$$

WSmax denotes the maximum walk step. In Equation (6), *dy* is the butterfly *j* walk step that Levy flight can consider.

$$dy = Levy(y^t{}_j) \tag{8}$$

Finally, the freshly formed butterfly with the best fitness is promoted to the next generation and replaced by its father; it is also eliminated to preserve population number.

Cauchy Distribution

The Cd is a separate updating technique that works in tandem with the migration and butterfly adjustment operators. This is the continuous probability distribution that has two parameters, x_0 and γ. x_0 is the location parameter, and γ is the scale parameter that defines the shape of the Cd. For instance, if a developed value is allotted to γ, the height of the peak of the Probability Density Function (PDF) will be smaller, and its width will be broader. On the other side, if a lesser value is allocated to γ, the height of the peak of the PDF will be higher, and its thickness will be narrower. The Cd's PDF may be described as follows.

$$f(x;x_0\,\gamma)=\frac{1}{\pi\gamma\left[1+\left(\frac{x-x_o}{\gamma}\right)^2\right]}=\frac{1}{\pi}\left[\frac{\gamma}{(x-x_o)^2+\gamma^2}\right] \quad (9)$$

The Cd's cumulative distribution function may also be described as follows.

$$F(x;x_0,\gamma)=\frac{1}{\pi}\arctan\left(\frac{x-x_0}{\gamma}\right)+\frac{1}{2} \quad (10)$$

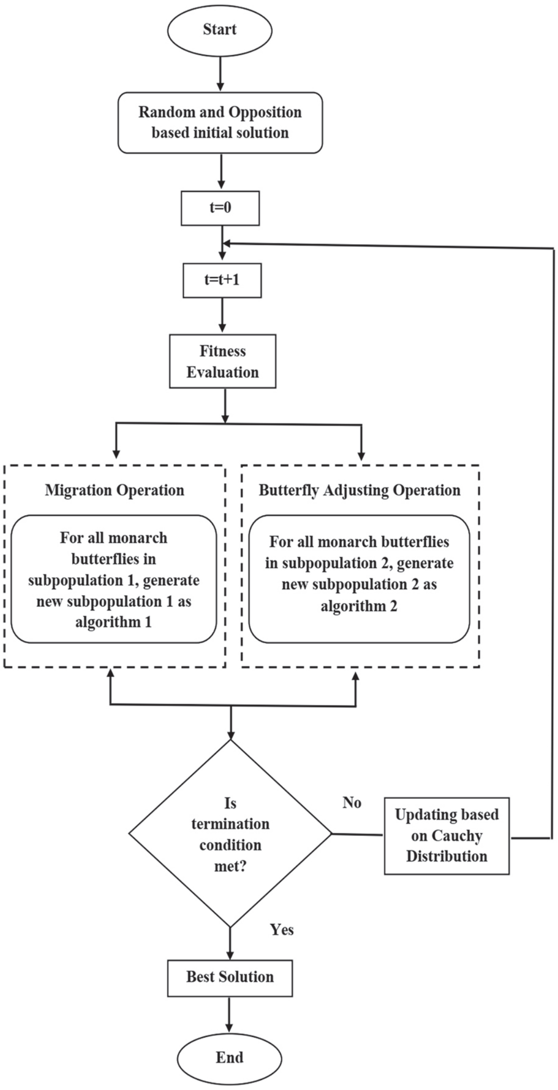

Fig. 2. Flow chart of Mutated Monarch Butterfly optimization

3.2.2 Comparison of Hyper parameters of all network structures

The default structure of ANN is comprised of one input layer, one hidden layer associated with ten neurons, and one output layer. Each neuron in the input layer is connected with the hidden layer neurons with random weight w11, w12...wij. Similarly, with the output layer. These initial random weights are adjusted based on the fed features. In Artificial Neural Networks, the Levenberg-Marquardt algorithm is most commonly used for training optimization, and the default transfer function is 'tansig'.

Table 1: Comparison of Hyper parameters of all network structures

	Updating Best solution in every iteration	Updating parameters in Land 1 and Land 2	Initial Population (NP monarch butterflies)	Training Algorithm	Transfer function of the neurons
Default ANN	No optimization techniques are employed for finding the optimum number of hidden layers and its associated neurons			trainlm	tansig
MBO configured ANN	No change	Migration Operator and Butterfly Adjusting operator	Random solution generation	trainlm	tansig
OMBO configured ANN	No change	Migration Operator and Butterfly Adjusting operator	Opposition-based solution generation	Trainlm	Tansig
CMBO configured ANN	The elements of best solution are further updated using Cauchy distribution	Migration Operator and Butterfly Adjusting operator	Random solution generation	trainlm	tansig
MMBO configured ANN	The elements of best solution are further updated using Cauchy distribution	Migration Operator and Butterfly Adjusting operator	Opposition-based solution generation	trainlm	tansig

In the basic MBO algorithm, local optima are easily reached, resulting in early convergence and poor performance. Using opposition-based learning (OBL) and the Cauchy distribution, this paper develops a novel MBO algorithm. Initially, OBL is used to create opposition-based populations from the original population. In opposition-based populations, the best individuals are selected and passed to the next generation, and this process effectively prevents the MBO from falling into a local optimum.

In this context, the optimal number of hidden layers and their associated neurons can be determined. If N is the number of hidden layer neurons generated randomly, then No is the opposition-based solution expressed in equation 1.

Secondly, Cauchy distribution is introduced to improve the migration and butterfly adjustment operators. In every iteration, it helps to update the best solution to improve the convergence rate.

4. RESULTS

Migration operator and butterfly adjusting operator in the MBO algorithm ensure monarch butterflies' search directions. In addition, the migration operator and the butterfly adjusting operator can be executed simultaneously. One of the advantages of MBO algorithm is its simplicity and ease of implementation. However, MBO algorithm drawback is poor optimization efficiency in solving complex optimization problems, which can be seen in the following aspect. The monarch butterflies r1 and r2 are randomly selected from Subpopulation1 and Subpopulation2, respectively. A worse monarch butterfly may be selected to share its features with a better one, leading to the population degenerating. This can be overcome by using Opposition-based Learning method. if the OBL approach is introduced into the initialization of the MBO algorithm, it can produce the opposition-based population. Then, the better individuals are selected to participate in the evolution from the union of the original populations and the opposition-based populations. Further every element in the best solution after every iteration is also updated by Cauchy distribution. Thus, these two operations increase the population diversity and expands the exploration scope of MBO. Further, it contributes to faster rate of convergence and better accuracy as well as sensitivity.

Configuring ANN via MMBO techniques accomplished 97.5% sensitivity for speaker recognition. Opposition based solution generation parallel with random solution generation and Cd function elevates the sensitivity over Oppositional based MBO, Cd based MBO and traditional MBO. The suggested MMBO creates an ANN with three hidden layers, each of which has 19, 23, and 23 neurons. The investigation shows the performance of involved techniques through diverse measures. It is obvious from the graphs that proposed approach having better performance over other techniques. The table 1 exhibits the ANN model configuration from different techniques. All at once, the employed optimization techniques in configuring ANN model show three-hidden layers. Though, the hidden layers are same for employed techniques change in respective hidden neurons impact effectively on proposed approach. The entire execution procedure took place on the MATLAB R2015a.

Table 2. ANN model configuration from different techniques

Techniques	Input	Hidden Layers	Neurons	Neurons	Neurons	Output
MBO-ANN	61	3	20	22	23	1
OMBO-ANN	61	3	20	25	21	1
CMBO-ANN	61	3	30	20	23	1
MMBO-ANN	61	3	19	23	23	1

The performance of the strategies used when configuring ANN for speaker recognition is exposed in Fig. 2. The results show that MBO's use of ANN configuration to forecast speaker voice recognition is superior to contest strategies.

True Positive (TP) - Recognised person’s voice correctly identified as recognised person

False Positive (FP) - Not-Recognised person’s voice incorrectly identified as recognised person

True Negative (TN) - Not-Recognised person’s voice correctly identified as not-recognised person

False Negative (FN) - Recognised person’s voice incorrectly identified as not-recognised person

The Fig.2 illustrates the performance of employed techniques w.r.t to real-time speaker voice database for recognition accuracy and Sensitivity standard measures. The performance of MMBO association in configuring ANN model demonstrates greater forecasting performance than other strategies implemented in this research, as seen in the following graphical depiction.

Accuracy: Accuracy is also used as a statistical measure to appropriately detect/reject the recognized / not-recognised person with respect to authenticate biometric characteristics.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{12}$$

Table 3. Performance measures of employed techniques

Techniques	TP	TN	FP	FN	Accuracy	Sensitivity
MBO-ANN	1.8	37.8	0.2	0.2	0.9900	0.900
OMBO-ANN	1.85	37.85	0.15	0.15	0.9925	0.925
CMBO-ANN	1.9	37.9	0.1	0.1	0.9950	0.950
MMBO-ANN	1.95	37.95	0.05	0.05	0.9975	0.975

Table 3 exhibits the performance measures for real-time from all emplyoed techniques along with TP, TN, FP and FN. The results exhibits that the performance of proposed technique is better than others.

MMBO configured ANN achieves accuracy of 99.75% that is 0.25% greater than CMBO configured ANN and 0.5% greater than OMBO configured MBO. Similarly MMBO configured ANN achieves sensitivity of 97.5% that is 2.5% greater than CMBO configured ANN and 5.0% greater than OMBO configured MBO.

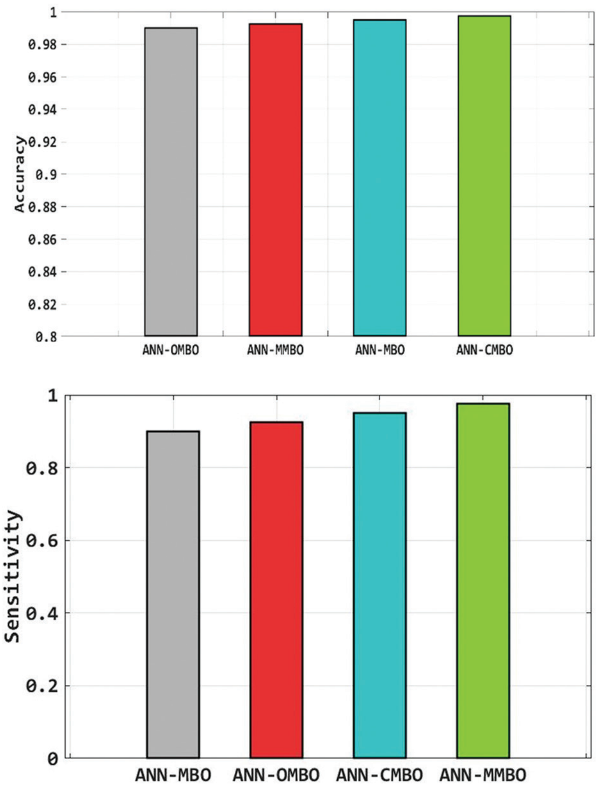

Fig. 3. Performance of Employed Techniques w.r.t standard measures

Converging Performance of the Employed Techniques

The following convergence graph shown in Fig. 4 signifies the performance of optimization techniques integrate with ANN for model configuration in real time speaker voice database. The performance of the used optimization association ANN approaches starts at the same point and gradually becomes exponential up to the 100th iteration, slowing down marginally after that. The proposed MMBO saturates at 400th iteration, which quite early over contest techniques; this is possible because of mutating both opposition and Cd strategy in traditional MBO.

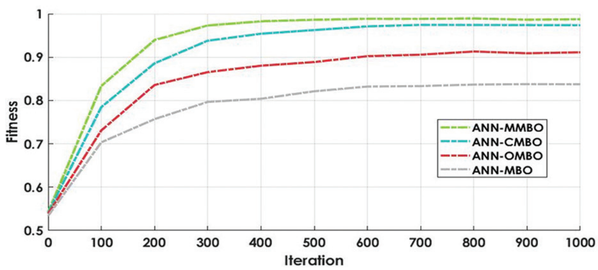

Fig. 4. Convergence graph

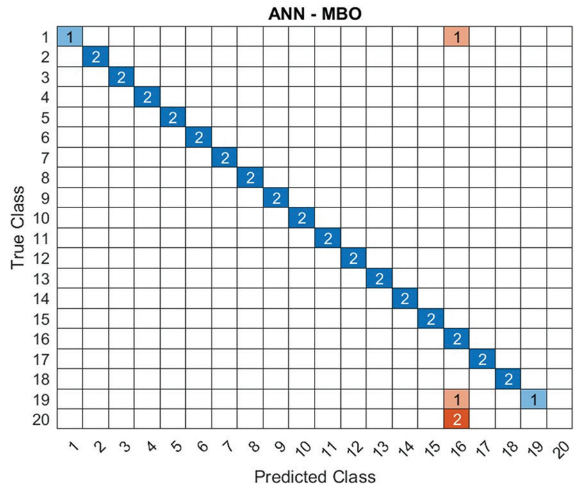

Fig. 5. Confusion matrix for real-time testing database by means of MBO – ANN

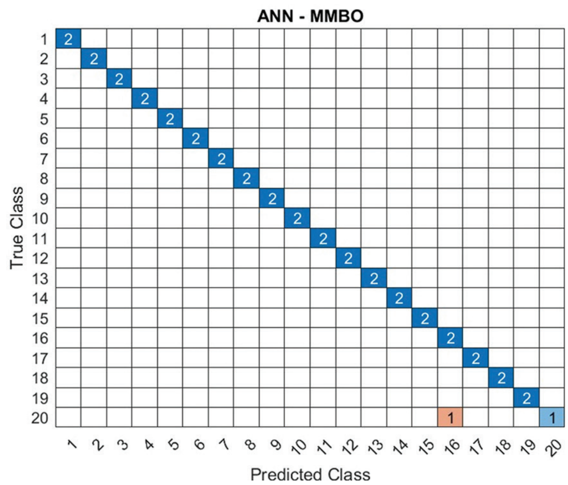

Fig. 6. Confusion matrix for real-time testing database by means of MMBO – ANN

Fig. 5 and Fig. 6 show the confusion charts for real-time speaker recognition using MBO-configured ANN and Mutated MBO-configured ANN, respectively. It is also evident from these figures that accuracy has improved.

5. CONCLUSION

Speaker recognition involves recognizing a person from a spoken word using a machine. It is possible to use speaker recognition systems either to recognize a specific individual or to validate the stated identification of that individual. In this study, voice data are gathered from cooperative office users with no unfavorable microphones. By using MMBO configured ANN models, we were able to recognize the speaker's voice with 97.5% sensitivity, which is superior to contest techniques. The hidden layer identifies characteristics in the input data and uses them to correlate an input with the appropriate output. An increase in hidden layers would complicate the model and lead to overfitting. A manual or trial-and-error approach to achieving those optimal configurations takes time and is time-consuming. Therefore, this research involves integrating optimization techniques and the superior performance is due to the modification of two important strategies, oppositional based solution generation and Cachy distribution in MBO. Research will focus in the future on large-scale speaker identification problems, which are quite challenging.

6. REFERENCES:

[1] N. N. An, N. Q. Thanh, Y. Liu, "Deep CNNs with self-attention for speaker identification", IEEE Access, Vol. 7, 2019, pp. 85327-85337.

[2] G. S. Morrison, F. H. Sahito, G. Jardine, D. Djokic, S. Clavet, S. Berghs, C. G. Dorny, "INTERPOL survey of the use of speaker identification by law enforcement agencies", Forensic science international, Vol. 263, 2016, pp. 92-100.

[3] M. G. Gomar, "System and method for speaker recognition on mobile devices", U.S. Patent 9,042,867, issued May 26, 2015.

[4] L. Chen, Y. Liu, W. Xiao, Y. Wang, H. Xie, "Speaker-GAN: Speaker identification with conditional generative adversarial network", Neurocomputing, Vol. 418, 2020, pp. 211-220.

[5] R. Jahangir, Y. W. Teh, N. A. Memon, G. Mujtaba, M. Zareei, U. Ishtiaq, M. Z. Akhtar, I. Ali, "Text-independent speaker identification through feature fusion and deep neural network", IEEE Access, Vol. 8, 2020, pp. 32187-32202.

[6] R. Jahangir, Y. W. Teh, H. F. Nweke, G. Mujtaba, M. Ali Al-Garadi, I. Ali, "Speaker identification through artificial intelligence techniques: A comprehensive review and research challenges", Expert Systems with Applications, Vol. 171, 2021, p. 114591.

[7] S. Tirumala, Sremath, S. R. Shahamiri, A. S. Garhwal, R. Wang, "Speaker identification features extraction methods: A systematic review", Expert Systems with Applications, Vol. 90, 2017, pp. 250-271.

[8] N. Lavan, A. M. Burton, S. K. Scott, C. McGettigan, "Flexible voices: Identity perception from variable vocal signals", Psychonomic bulletin & review, Vol. 26, No. 1, 2019, pp. 90-102.

[9] Z. Qawaqneh, A. A. Mallouh, B. D. Barkana, "Deep neural network framework and transformed MFCCs for speaker's age and gender classification", Knowledge-Based Systems, Vol. 115, 2017, pp. 5-14.

[10] N. Almaadeed, A. Aggoun, A. Amira, "Text-independent speaker identification using vowel formants", Journal of Signal Processing Systems, Vol. 82, No. 3, 2016, pp. 345-356.

[11] Z. Ma, H. Yu, Z.-H. Tan, J. Guo, "Text-independent speaker identification using the histogram transform model", IEEE Access, Vol. 4, 2016, pp. 9733-9739.

[12] J.-Y. Lai, S.-L. Wang, A. Liew, X. Shi, "Visual speaker identification and authentication by joint spatiotemporal sparse coding and hierarchical pooling", Information Sciences, Vol. 373, 2016, pp. 219-232.

[13] Y. H. Jung, S. K. Hong, H. S. Wang, J. H. Han, T. X. Pham, H. Park, J. Kim, S. Kang, C. D. Yoo, K. J. Lee, "Flexible piezoelectric acoustic sensors and machine learning for speech processing", Advanced Materials, Vol. 32, No. 35, 2020, p. 1904020.

[14] R. Chakroun, M. Frikha, "New approach for short utterance speaker identification", IET Signal Processing, Vol. 12, No. 7, 2018, pp. 873-880.

[15] N. Almaadeed, A. Aggoun, A. Amira, "Speaker identification using multimodal neural networks and wavelet analysis", IET Biometrics, Vol. 4, No. 1, 2015, pp. 18-28.

[16] N. Saxena, D. Varshney, "Smart Home Security Solutions using Facial Authentication and Speaker Recognition through Artificial Neural Networks", International Journal of Cognitive Computing in Engineering, Vol. 2, 2021, pp. 154-164.

[17] I. Aljarah, H. Faris, S. Mirjalili, "Optimizing connection weights in neural networks using the whale optimization algorithm", Soft Computing, Vol. 22, No. 1, 2018, pp. 1-15.

[18] D. Devikanniga, R. J. S. Raj, "Classification of osteoporosis by artificial neural network based on monarch butterfly optimisation algorithm", Healthcare technology letters, Vol. 5, No. 2, 2018, pp. 70-75.

[19] P. Soltani, E. Hadavandi, "A monarch butterfly optimization-based neural network simulator for prediction of siro-spun yarn tenacity", Soft Computing, Vol. 23, No. 20, 2019, pp. 10521-10535.

[20] H. Faris, I. Aljarah, S. Mirjalili, "Improved monarch butterfly optimization for unconstrained global search and neural network training", Applied Intelligence, Vol. 48, No. 2, 2018, pp. 445-464.

[21] D. Yang, X. Wang, X. Tian, Y. Zhang, “Improving monarch butterfly optimization through simulated annealing strategy”, Journal of Ambient Intelligence and Humanized Computing, 2020, pp. 1-12.

[22] X. Yu, W. Xu, C. Li, “Opposition-based learning grey wolf optimizer for global optimization”, Knowledge-Based Systems, Vol. 226, 2021, p. 107139.

[23] S. Mahdavi, S. Rahnamayan, K. Deb, “Opposition based learning: A literature review”, Swarm and Evolutionary Computation, Vol. 39, 2018, 1-23.

[24] W. Wang, L. Xu, K. Chau, D. Xu, “Yin-Yang firefly algorithm based on dimensionally Cauchy mutation”, Expert Systems with Applications, Vol. 150, 2020, p. 113216.

[25] K. Daqrouq, T. A. Tutunji, “Speaker identification using vowels features through a combined method of formants, wavelets, neural network classifiers”, Applied Soft Computing, Vol. 27, 2015, pp. 231-239.

[26] O.S. Faragallah, "Robust noise MKMFCC–SVM automatic speaker identification”, International Journal of Speech Technology, Vol. 21, No. 2, 2018, pp. 185-192.

[27] C. Chen, W. Wang, Y. He, J. Han, “A bilevel framework for joint optimization of session compensation and classification for speaker identification”, Digital Signal Processing, Vol. 89, 2019, pp. 104-115.

[28] V. de Abreu Campos, D. Guimarães Pedronette, “A framework for speaker retrieval and identification through unsupervised learning”, Computer Speech & Language, Vol. 58, 2019, pp. 153-174.

[29] S. Safavi, M. Russell, P. Jančovič, “Automatic speaker, age-group and gender identification from children’s speech”, Computer Speech & Language, Vol. 50, 2018, pp. 141-156.

[30] K. Devi, J., N. H. Singh, Khelchandra Thongam, “Automatic speaker recognition from speech signals using self-organizing feature map and hybrid neural network”, Microprocessors and Microsystems, Vol. 79, 2020, p. 103264.

[31] S. Biswas, S. S. Solanki, “Speaker recognition: an enhanced approach to identify singer voice using neural network”, International Journal of Speech Technology, Vol. 24, No. 1, 2021, pp. 9-21.

[32] G. G. Wang, S. Deb, Z. Cui, “Monarch butterfly optimization”, Neural computing and applications, Vol. 31, No. 7, 2019, 1995-2014.

[33] F. Chakraborty, P. K. Roy, D. Nandi, “Oppositional elephant herding optimization with dynamic Cauchy mutation for multilevel image thresholding”, Evolutionary Intelligence, Vol. 12, No. 3, 2019, pp. 445-467.