SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I3P118

Volume 11 Issue 3, 220-230, March 2024
© 2024 Seventh Sense Research Group®

Original Article

Measuring the Accuracy of A Voiceprint Analysis System Designed by Applying the Euclidean Distance Function and Genetic Algorithm

Asmaa Barakat[1], Abu Naqra[2], Abdul Rahman Hussian[3]

[1,3]*Faculty of Electrical and Electronic Engineering, University of Idlib, Idlib, Syria.*
[2]*Faculty of Electrical and Electronic Engineering, Istanbul University, Istanbul, Turkey.*

Corresponding Author : abdul.hussian@istanbul.edu.tr

Received: 11 January 2024 Revised: 09 February 2024 Accepted: 10 March 2024 Published: 25 March 2024

Abstract - *This research aims to measure the accuracy of the work of the voiceprint analysis system. The system comprises three stages: (i) recording voice, deleting noise, and extracting the voiceprint, (ii) establishing the database, and (iii) comparing the data and decision-making process. The process of deleting the noise and extracting the voiceprint, in which noise deletion is the biggest challenge, is in the first stage. Next, the voice is analyzed by applying the MFCC algorithm, and then a statistical equation is utilized to extract the voiceprint. Creating a database in which the voiceprint samples are saved and comparing and making a decision by applying the Euclidean distance function and the genetic algorithm, respectively. The test results showed speakers' recognition ratios among the user groups (10, 20, 30, 40), by applying the Euclidean distance function, are (93%, 89.5%, 82.83%, and 73.37%) respectively. The distinction was improved by adding the genetic algorithm to the Euclidean distance function for the same number of users. The results were as follows (94%, 90.75%, 83.83%, and 74.87%), respectively. The average time for voice analysis and voiceprint extraction was (3.183, 3.174, 3.171, and 3.169 sec.); the average time for testing (0.00807, 0.00808, 0.0082, and 0.0258 sec.) by applying the Euclidean distance function; and the average time for testing (0.00615, 0.023711, 0.020747, and 0.022438 sec.) by applying the Euclidean distance function and the genetic algorithm, and thus speeding up the testing and decision-making process is achieved.*

Keywords - *Voiceprint, MFCC algorithm, Euclidean distance, Genetic Algorithm,CNN, ANN.*

1. Introduction

Stealing passwords is the biggest concern for owners of important facilities, so advanced protection systems are required to secure the desired protection for those facilities and buildings. Hence, passwords based on biometrics have appeared, including the voiceprint [1].

It is impossible to steal a voice print, and if the voice is imitated, the devices used will detect identity theft. Much research and studies have been conducted in this field, and voiceprint systems have been built to identify the speaker (User ID verification), and multiple algorithms are used in the stage of analyzing the voice and extracting its characteristics. Some of these use the spectral field in the analysis of voice, using a popular algorithm for voice signal spectroscopy, and some depend on prosodic features, such as obtaining the basic frequency or energy by applying a form of equation that gives the voice signal energy [2].

These algorithms, which rely on spectro-domain analysis, include the Mel Frequency Cepstral Coefficients MFCC, which is the most widely in this field [3], the Linear Frequency Cepstrum Coefficients (LFCC) [4], the Linear Predictive Coding (LPC) [5], the Linear Prediction Cepstral Coefficients (LPCC) [6], the Mel-frequency Cepstral Coefficients MFCC [7], the Gamma-tone Frequency Cepstral Coefficients (GFCC) [7], and the Power Normalized Cepstral Coefficients (PNCC) [8].

Each algorithm relies on the application of a Fourier transform depending on the type of algorithm; thus, a transform is made from the time to frequency domain, which contains more information about the voice signal. Then, the voice signal is passed through filters in some algorithms, and some of these algorithms have different types of filters.

This spectroscopy process simulates the work of the human ear in analyzing sound. After that, the filter output is aggregated by applying a specific equation. Each algorithm has its own equation, and the assembly process through applying a certain equation simulates the work of the human brain. The spectroscopy involves the analysis and grouping of this voice signal, and from this spectroscopy, a matrix representing the characteristics of the voice signal is obtained.

Voice characteristics are extracted using several methods, including Principal Component Analysis (PCA) and genetic algorithm [9], or by applying the classifier used in the voice recognition and decision-making process directly 1 and these studies are applied to ready-made databases downloaded from the internet [10, 11] or are recorded by the researcher on a specified number of people [1]. The database is divided into two parts: training [11] and testing the proposed system [11].

The final stage is recognition and decision-making, using several classifiers. First, conventional classifiers [2], including Support Vector Machine (SVM) [12], Hidden Markov Model (HMM) [13], Gaussian Mixture Model (GMM) [14], k-Nearest Neighbors Classifier [15], i-Vector [8], Artificial Neural Network (ANN) [16].

Second, Deep Learning classifiers [2] comprising Convolutional Neural Networks (CNN) [17], Deep Neural Networks (DNN) and Recurrent Neural Networks (RNN) [18], and Long Short-Term Memory (LSTM) [19]. Finally, technological optimization classifiers based on Deep Learning [2], including automatic encoding [20], and multitask learning [21].

Each of these classifiers has a different way of recognizing sounds. Given that time is a critical factor in speaker identification. Despite the importance of the voice signal and its varied uses in the area of information security, there are some problems with the time a user waits before being identified and the need not to allow the wrong people to enter, thereby jeopardizing the security of the system.

Hence, in this paper, we provided a statistical equation is provided to achieve a high speed of voiceprint extraction, reaching fractions of a second to recognize the speaker, and several thresholds have been experimented with to increase system accuracy and reduce the false acceptance rate to zero.

2. Materials and Methods

A voice signal is an analog signal with amplitude, frequency, and phase, which is continuous over time, and because it is large, it cannot be fully processed but is divided into equal sections called time frames [22].

2.1. Voice Signal Analysis and Voiceprint Extraction

The voice is recorded and then analyzed by the following:

2.1.1. Noise Purification

The biggest challenge in processing voice signals is noise disposal; the voice signal is subject to two types of noise: thermal noise, which is the noise caused by the devices used, such as internal noise in the computer, microphone, and cable between the computer and the microphone, and external noise, such as noise from pedestrians and cars near the place where the system is used [23]. To eliminate the noise, we do the following:

Thermal Noise Deletion

Recording without speaking in front of the microphone is applied. Hence, only the noise signal is recorded. Then, the arithmetic mean of the amplitude of the recorded noise signal is calculated by applying the following equation:

$$Continu\_value = \frac{1}{N}\sum_{i=1}^{N} X_i \tag{1}$$

Where,
Xi : Voice samples.
N : Number of voice samples.

The value of the continuous signal representing thermal noise is obtained and then subtracted from the value of the voice signal samples to get the original voice signal [23].

Deleting Silence Periods

When a statement during speech is recorded, and there are silence periods between words, those intervals are deleted and thus reduce the amount of data processed and speed up the extraction of the voiceprint [23].

To determine the threshold for deleting silence periods and to see if the time frame of the voice signal contains only a voice or noise signal, the amplitude equation to the recorded signal without speaking to the microphone is applied by calculating the arithmetic mean of the voice sample values but in absolute terms:

$$Amp_{Avrg} = \frac{1}{N} \cdot \sum_{i=1}^{N} |X_i| \tag{2}$$

Where,
Xi : Voice samples.
N : Number of voice samples.

A phrase in front of the microphone is recorded and then divided into equal frames. If the maximum value in the frame is greater than this threshold, the time frame is retained, but if the greatest value within the frame is smaller than this threshold, the frame is deleted, which is a part of the voice signal that contains noise only and does not contain spoken speech [23].

2.1.2. Voice Signal Analysis

After purifying the signal of noise, we analyze it by applying the following mathematical process:

Applying the MFCC Algorithm

This algorithm, which is the most famous one in this field, is applied by the following steps:

- Dividing the voice signal into frames, the time of each frame (10 m.s). Because the signal is variable with time, it is only processed by dividing it into equal periods, and it is considered somewhat stable within these frames to be able to apply equations to it [22].
- Hitting the signal by the Hamming Window to focus the value of the data in the center [1].
- Applying FFT to move from the time domain to the frequency.
- Applying the following approximate equation to convert each obtained frequency to a Mel frequency [22]:

$$Mel(f) = 2595 * log10\left(1 + \frac{f}{700}\right) \qquad (3)$$

Where, f - frequency of the voice signal.

- The Discrete Cosine Transformation (DCT) is used to reconvert the spectrum of the logarithmic Mel field to the time domain by using the following equation:

$$MFCC = \sum_{K=1}^{20} Xk\ .COS\ \frac{\pi\ .i(k-0.5)}{20}, i = 1, 2, 3, \dots p \qquad (4)$$

Where,
20 : The number of filters we have selected.
X_k : Frame value.
P : Number of fixed MFCC coefficients for the analyzed signal, and in this research, it equals 14.

Thus, the harmonic matrix is obtained, which is the number of lines by the number of frames and the number of columns by the number of analyzed coefficients [3].

Applying the Dynamic MFCC Equation

The dynamic MFCC coefficients delta is obtained by applying the following equation:

$$Delta = \frac{\sum_{\theta=1}^{\Theta} \theta(C_{i+\theta} - C_{i-\theta})}{2\sum_{\theta=1}^{\Theta} \theta^2} \qquad (5)$$

Where,
Ɵ : Dynamic MFCC coefficient.
Θ : The width of the delta window, and here its value is equal to [4].
C_i : Constant MFCC coefficient.

Also, applying the same equation to the MFCC dynamic coefficients delta gives the dynamic MFCC coefficients delta-delta [24].

Extracting the Voiceprint Vector

By applying the MFCC algorithm to the user's voice signal, (14) coefficients of MFCC are chosen. After several coefficients were used, it was found that the number of coefficients (14) worked. After applying the dynamic MFCC delta equation to the same acoustic signal, another (14) coefficient was chosen, dynamic MFCC delta-delta to the same acoustic signal was applied, and (14) coefficients were chosen.

Thus, a column matrix equal to the number of coefficients of the MFCC, MFCC dynamic delta, and MFCC dynamic delta-delta is obtained, and the number of lines in this matrix is equal to the number of frames obtained from the recording of the voice signal. Such a matrix is obtained in each recording. Matching between matrices is difficult and takes a long time, so there is a need to shorten this data, but without making it lose its features so the goal was to achieve the following:

- The possibility of grouping the voiceprint vectors of the same person with one set.
- The possibility of recognizing the voiceprints of the same person from those of other people.

The resulting matrix is reduced to a single line matrix, which has the same number of columns (14+14+14), which is called a voiceprint vector, by applying the empirically obtained statistical equation by dividing the arithmetic mean by the standard deviation for each column of the matrix:

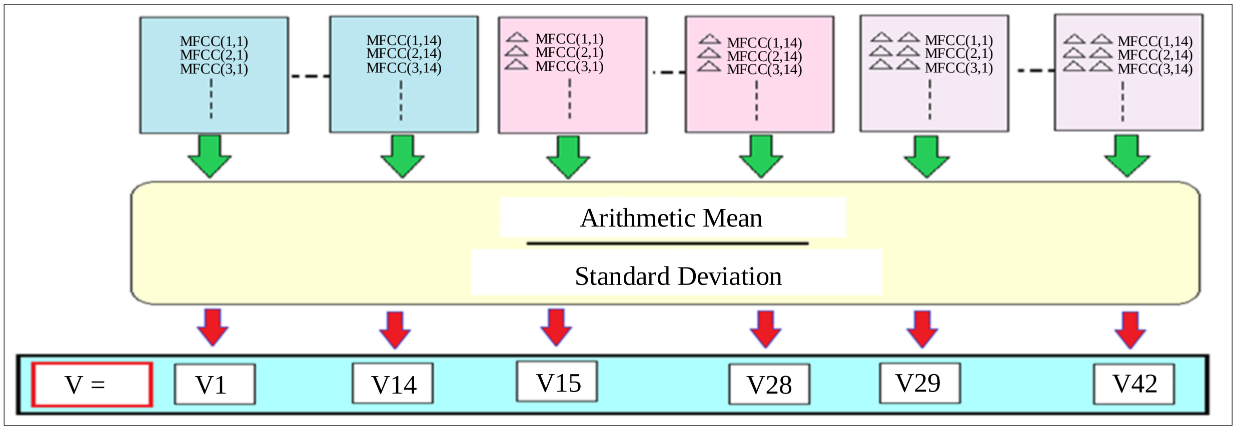

Fig. 1 Voiceprint vector extraction chart

Where,
MFCC(i,j) : MFCC coefficient (j) for the frame (i).
i : Frame number.
j : Column number.
Vj : The value of coefficient (j).
V : A vector representing the voiceprint, a line and (42) columns for each recorded voice.

By applying the statistical equation, the problem of not being able to specify the exact time of saying "Assalamu Alaikum" (Hello), representing a difference in the number of frames produced from one recording to another, was overcome, leading to a difference in the number of vectors obtained from each recording. Hence, using this equation reduced the amount of data used, which made the system respond more quickly.

3. Database Creation

The voices of the people whose voice prints would be extracted were recorded in a noisy environment to test the proposed system. The voices of 40 people, (20) males and (20) females, were recorded between the ages of (20 to 60). Each of them recorded a phrase in Arabic: "Open the door", and each voice was recorded (20) times in two groups: a training group and a test group, thus bringing the number of files to 400 in the training group and 400 in the test group, and the database included 800 voices.

The recording was done under appropriate conditions and at a sample rate equal to (44100). The recording was done in similar conditions for all people, and the voice was recorded by speaking in front of the microphone. Then, a signal was recorded without speaking in front of the microphone, so the noise signal was recorded, and the thermal noise value was calculated and deleted. The silence periods were deleted, so the voice signal was ready for analysis. The MFCC algorithm was applied to obtain harmonics in each frame, and by applying the statistical equation reached in the experiment, the voiceprint vector was produced.

4. Recognition and Decision-Making

For recognition and decision-making, the Euclidean distance function is applied between the voiceprint vectors, which is given by the equation [25]:

$$Dji = \frac{1}{p}\sqrt{\sum_{k=1}^{p} \left(ref(C,k) - test(h,k)\right)^{2}} \quad (6)$$

Where,
Dji : The difference between two vectors of the voiceprint.
ref : A vector of the database.
test : Tested voiceprint vector.
K : Column number.
h, C : Line number.

However, the human voice signal is affected by the psychological, mood, and pathological state, which is reflected in his voice's print. To reduce the impact of these emotions on the person's voiceprint, the genetic algorithm is applied to increase the percentage of recognition of the person while maintaining the reliability of this recognition. The system is made as follows [26]:

1. When a person wants to enter, he records the same phrase that was recorded in the database, "Open the door", in front of the microphone, then the voice is analyzed, and the person's voiceprint vector is extracted.
2. The voiceprint is compared with the database, and the difference between it and all the voiceprints in the database is recorded using the Euclidean distance function. The results are arranged from the closest voice to the farthest one.
3. If the difference between it and the closest voice in the database is less than the proposed threshold that was reached experimentally, the person is allowed to enter, but if the difference is greater than the threshold, the steps of the genetic algorithm are applied.

   Each voiceprint vector is considered a chromosome, and the chromosome consists of several genes. The value of each coefficient in the voiceprint vector represents a specific gene; that is, the number of chromosomes in the database is equal to the number of lines of the database matrix (the number of people × the number of voices of each person), and the number of genes in each chromosome is equal to the number of coefficients in each voiceprint vector, i.e. (42) genes, and the registered voiceprint vector of the person who wants to enter is a chromosome. The chromosome represents an individual and consists of (42) genes as well.

   The voiceprint vectors in the database are considered to be first-generation individuals (i.e., parents). The difference between the voiceprint vectors is calculated by applying the Euclidean relationship. The threshold is chosen when the difference between the voiceprint samples of the same person is smaller than the threshold, and the difference between the voiceprint samples of different people is greater than the threshold.

4. The value of the Euclidean difference is calculated between the recorded voiceprint vector and all voiceprint vectors in the database (i.e., first-generation), and the difference is stored in the difference column (Column 43 in the database matrix).
5. Based on the value of this difference and in ascending order, the generation members are rearranged; the vectors are arranged from the vector closest to the voiceprint to the farthest one.
6. A specific number of parents (primary chromosomes) are copied from the beginning of the parent generation matrix

to the new generation to preserve the best characteristics, i.e. to preserve the voiceprint vectors closest to the recorded voiceprint vector (the best parents).

7. A matrix with fewer lines is obtained from the database representing the best parents, and mating occurs between two parents (chromosomes - two lines of the matrix). The mating between two chromosomes produces two new chromosomes representing the children. The generation of children is added to the matrix of stronger parents, so the chromosomes of the parents with the least difference (the best chromosomes) and the children's chromosomes resulting from the mating process between these chromosomes are produced.
8. The concept of mutation to several new chromosomes is applied by generating random values. In this research, the Rand function in the Matlab language is relied on to generate these random values, so the chromosome and the gene in which the mutation occurs are randomly chosen. The mutation exclusively occurs in the chromosomes of the children.
9. The difference is calculated for new individuals (stronger parents, children without a mutation, and children with a mutation) by calculating the Euclidean difference of the lowest distance between the new individuals and the chromosome, which represents the recorded voice of the person who wants to enter, and by storing the value of this difference in the difference column (Column 43 of the database matrix) in which all the different values for the new generation will be stored. The smallest difference value is the optimal value.
10. If the value of the optimal value is less than the threshold, the person is allowed to enter, but if it is greater than the threshold, steps are repeated from 3 to 10 with a specified number of times (number of generations). At the end of the application of each generation, the optimal value is compared with the threshold; if it is lower, the decision is made to accept the entrance. Still, the transition to a new generation is made if it is greater. If the number of generations ends and the optimal difference value is less than the threshold, the person is allowed to enter. Still, if the optimal difference value is greater than the threshold, the person is not allowed to enter.

5. Results and Discussion

5.1. Results

The following results are obtained by both applying the Euclidean distance function between the training matrix and the test matrix per person, calculating the recognition ratio and comparing the training matrix per person with the total matrix, which contains the training matrix for the admitted persons, except for the same person, to test the system, calculate its accuracy ratio, and determine recognition time by applying different thresholds and testing the voice.

Whereas:

- A: It is the set of voiceprints in which the difference (optimal value) between any voiceprint of the test group and all the voiceprints of the training group is less than the threshold by applying the Euclidean distance function only for the same person.
- Q: False refusals in which persons allowed to enter are incorrectly denied.
- B: It is the set of voiceprints in which the difference (optimal value) between any voiceprint of the test group and all the voiceprints of the training group is less than the threshold for the same person. The difference is calculated by applying the Euclidean Distance function, and the results are optimized using the genetic algorithm.
- R: False refusals in which people are allowed to enter are incorrectly denied by applying the Euclidean distance function and the genetic algorithm.
- C: It is the set of voiceprints of the same person, in which the difference (optimal value) between any voiceprint in the test group and all the voiceprints of the total database is without the person's voiceprints to determine the reliability of the system and thus determine the value of the falsely accepted entrance, i.e. they are cases of incorrectly accepted entrance for persons who are not allowed to enter.
- D: The percentage of voices accepted before using the genetic algorithm for each person.
- G: The percentage of false rejection in each person's voice set before applying the genetic algorithm.
- E: The percentage of acceptable voices by applying a genetic algorithm for each person.
- K: The percentage of false rejection in each person's voice group.
- F: The percentage of false access in each person's voice set, with the genetic algorithm applied.
- T1: Arithmetic mean of time required to analyze the voice and extract the voiceprint vector.
- T2: Time required to compare the voice with the voices in the database and decide whether to accept or reject by the application of the Euclidean distance function.
- T3: Time required to analyze and test voice, i.e. voice analysis time and the time of test and decision making by applying the Euclidean distance function.
- T4: Time required to compare the voice with the voices in the database and decide whether to accept or reject entrance by applying the Euclidean distance function and the genetic algorithm.
- T5: Time required for voice analysis and testing, i.e., voice analysis time as well as test time and decision-making by applying the Euclidean distance function and genetic algorithm.
- a: Euclidean distance threshold, which is a value that is determined empirically. According to this threshold, the decision is made to accept or reject entrance.

Table 1. Analysis results of ten users at Euclidean threshold a≤0.12

Gender	Number of Users	A	Q	B	R	C	D (%)	G (%)	E (%)	K (%)	F (%)	T1 (sec)	T2 (sec)	T3 (sec)	T4 (sec)	T5 (sec)
Female	1	18	2	18	2	0	90	10	90	10	0	3.2203	0.0088	3.2291	0.0123	3.233
	2	17	3	18	2	0	85	15	90	10	0	3.2172	0.0037	3.2209	0.005	3.222
	3	20	0	20	0	0	100	0	100	0	0	3.1773	0.0097	3.187	0.0067	3.184
	4	20	0	20	0	0	100	0	100	0	0	3.1465	0.0047	3.1512	0.0048	3.151
	5	20	0	20	0	0	100	0	100	0	0	3.2019	0.0037	3.2056	0.0043	3.206
Male	6	18	2	19	1	0	90	10	95	5	0	3.1437	0.0273	3.171	0.0068	3.151
	7	20	0	20	0	0	100	0	100	0	0	3.143	0.0056	3.1486	0.0043	3.147
	8	19	1	19	1	0	95	5	95	5	0	3.1457	0.0033	3.149	0.0073	3.153
	9	17	3	17	3	0	85	15	85	15	0	3.2148	0.0108	3.1885	0.0043	3.219
	10	17	3	17	3	0	85	15	85	15	0	3.2213	0.0031	3.1654	0.0057	3.227

Table 2. Analysis results for twenty users at the Euclidean threshold a≤0.12

Gender	Number of Users	A	Q	B	R	C	D (%)	G (%)	E (%)	K (%)	F (%)	T1 (sec)	T2 (sec)	T3 (sec)	T4 (sec)	T5 (sec)
Female	1	18	2	18	2	0	90	10	90	10	0	3.2203	0.1515	3.3718	0.0119	3.2322
	2	17	3	18	2	0	85	15	90	10	0	3.2172	0.0126	3.2298	0.0165	3.2337
	3	20	0	20	0	0	100	0	100	0	0	3.1773	0.0067	3.184	0.0087	3.186
	4	20	0	20	0	0	100	0	100	0	0	3.1465	0.006	3.1525	0.0372	3.1837
	5	20	0	20	0	0	100	0	100	0	0	3.2019	0.0057	3.2076	0.0132	3.2151
	6	19	1	19	1	0	95	5	95	5	0	3.1419	0.006	3.1479	0.0203	3.1622
	7	20	0	20	0	0	100	0	100	0	0	3.1423	0.0062	3.1485	0.0144	3.1567
	8	18	2	18	2	0	90	10	90	10	0	3.1376	0.0052	3.1428	0.0192	3.1568
	9	17	3	18	2	0	85	15	90	10	0	3.1765	0.0101	3.1866	0.0098	3.1863
	10	19	1	19	1	3	95	5	95	5	0	3.1728	0.0053	3.1781	0.0097	3.1825
Male	11	18	2	19	1	0	90	10	95	5	0	3.1437	0.0069	3.1506	0.0174	3.1611
	12	20	0	20	0	0	100	0	100	0	0	3.143	0.0174	3.1604	0.0286	3.1716
	13	19	1	19	1	0	95	5	95	5	0	3.1457	0.012	3.1577	0.0571	3.2028
	14	17	3	17	3	0	85	15	85	15	0	3.2148	0.0062	3.221	0.0138	3.2286
	15	17	3	17	3	0	85	15	85	15	0	3.2213	0.0075	3.2288	0.0205	3.2418
	16	17	3	18	2	0	85	15	90	10	0	3.2268	0.0068	3.2336	0.0086	3.2354
	17	16	4	16	4	0	80	20	80	20	0	3.1674	0.0068	3.1742	0.0147	3.1821
	18	20	0	20	0	0	100	0	100	0	0	3.1777	0.0029	3.1806	0.0113	3.189
	19	20	0	20	0	0	100	0	100	0	0	3.1652	0.006	3.1712	0.016	3.1812
	20	19	1	19	1	0	95	5	95	5	0	3.1406	0.005	3.1456	0.0247	3.1653

Table 3. Analysis results for twenty users at the Euclidean threshold a≤0.115

Gender	Number of Users	A	Q	B	R	C	D (%)	G (%)	E (%)	K (%)	F (%)	T1 (sec)	T2 (sec)	T3 (sec)	T4 (sec)	T5 (sec)
Female	1	18	2	18	2	0	90	10	90	10	0	3.2203	0.0156	3.2359	0.0286	3.2489
	2	15	5	16	4	0	75	25	80	20	0	3.2172	0.0152	3.2324	0.0144	3.2316
	3	17	3	18	2	0	85	15	90	10	0	3.1773	0.0062	3.1835	0.0211	3.1984
	4	20	0	20	0	0	100	0	100	0	0	3.1465	0.0067	3.1532	0.011	3.1575
	5	20	0	20	0	0	100	0	100	0	0	3.2019	0.0061	3.1995	0.0317	3.2336
	6	19	1	19	1	0	95	5	95	5	0	3.1419	0.012	3.1539	0.0124	3.1543
	7	20	0	20	0	0	100	0	100	0	0	3.1423	0.0051	3.1474	0.0162	3.1585
	8	15	5	16	4	0	75	25	80	20	0	3.1376	0.0058	3.1434	0.0105	3.1481
	9	17	3	17	3	0	85	15	85	15	0	3.1765	0.0064	3.1829	0.0385	3.215
	10	17	3	17	3	0	85	15	85	15	0	3.1728	0.0147	3.1875	0.014	3.1868
Male	11	17	3	18	2	0	85	15	90	10	0	3.1437	0.0063	3.15	0.0165	3.1602
	12	20	0	20	0	0	100	0	100	0	0	3.143	0.007	3.15	0.1164	3.2594
	13	18	2	18	2	0	90	10	90	10	0	3.1457	0.0065	3.1522	0.0182	3.1639
	14	17	3	17	3	0	85	15	85	15	0	3.2148	0.0057	3.2205	0.0154	3.2302
	15	17	3	17	3	0	85	15	85	15	0	3.2213	0.0062	3.2275	0.0169	3.2382
	16	16	4	17	3	0	80	20	85	15	0	3.2268	0.0069	3.2337	0.0342	3.261
	17	16	4	16	4	0	80	20	80	20	0	3.1674	0.005	3.1724	0.01	3.1774
	18	20	0	20	0	0	100	0	100	0	0	3.1777	0.0063	3.184	0.0164	3.1941
	19	20	0	20	0	0	100	0	100	0	0	3.1652	0.0117	3.1769	0.0081	3.1652
	20	19	1	19	1	0	95	5	95	5	0	3.1406	0.0062	3.1468	0.0125	3.1487

Table 4. Arithmetic mean of (10-20-30-40) used for previous values at the thresholds taken

User Numbers	Euclidean Distance Threshold, a	Arithmetic Mean for A	Arithmetic Mean for Q	Arithmetic Mean for B	Arithmetic Mean for R	Arithmetic Mean for C	Arithmetic Mean for D (%)	Arithmetic Mean for G (%)	Arithmetic Mean for E (%)	Arithmetic Mean for K (%)	Arithmetic Mean for F (%)
10	0.12	18.6	1.4	18.8	1.2	0	93	7	94	6	0
20	0.12	18.55	1.45	18.75	1.25	0.15	92.75	7.25	93.75	6.25	0. 75
	0.115	17.9	2.1	18.15	1.85	0	89.50	10.50	90.75	9.25	0
30	0.12	18.83	1.666	18.96	1.03	0.6	94.16	5.83	94.83	5.16	3
	0.115	18.3	1.7	18.46	1.53	0.33	91.50	8.50	92.33	7. 66	1.60
	0.11	17.63	2.366	17.8	2.2	0.23	88.16	11.83	89	11	1.10
	0.105	16.56	3.433	16.76	3.23	0	82.83	17.17	83.83	16.16	0
40	0.12	19.07	0.925	19.18	0.83	1.12	95.37	4.63	95.87	4.12	5.60

	0.115	18.62	1.375	18.75	1.25	0.72	93.12	6.88	93.75	6.25	3.50
	0.11	18.1	1.9	18.23	1.78	0.4	90.50	9.50	91.13	8.88	2
	0.105	17.25	2.75	17.4	2.6	0.12	86.25	13.75	87	13	0.70
	0.1	14.675	5.325	14.98	5.03	0	73.37	26.63	74.88	25.125	0

Table 5. Arithmetic mean of sound analysis and test time

User Numbers	Euclidean Distance Threshold, a	Arithmetic Mean for T1	Arithmetic Mean for T2	Arithmetic Mean for T3	Arithmetic Mean for T4	Arithmetic Mean for T5
10	0.12	3.1832	0.00807	3.191	0.01	3.18932
20	0.12	3.174	0.01464	3.189	0.02	3.19271
	0.115	3.174	0.00808	3.182	0.02	3.19655
30	0.12	3.1717	0.01576	3.187	0.03	3.19884
	0.115	3.1717	0.01094	3.183	0.03	3.19684
	0.11	3.1717	0.01059	3.182	0.02	3.18979
	0.105	3.1717	0.00824	3.18	0.02	3.19219
40	0.12	3.1699	0.01739	3.187	0.02	3.19372
	0.115	3.1699	0.023	3.193	0.02	3.19246
	0.11	3.1699	0.02226	3.192	0.02	3.19265
	0.105	3.1699	0.02456	3.194	0.02	3.19416
	0.1	3.1697	0.02589	3.196	0.02	3.19231

To reduce the system's error rate, the experimentally reached threshold is reduced to increase system reliability, thus decreasing the number of voices allowed to enter as the threshold is reduced.

5.2. Discussion

The optimal conventional threshold achieves a false acceptance rate of zero; that is, no person who is not admitted is allowed to enter. Through previous experiments and recordings, it is concluded that the best thresholds reached were (0. 120, 0.115, 0.105, 0.100) for the groups of users (10, 20, 30, 40), respectively. Consequently, as the number of users increased, the threshold was lowered to maintain a zero false acceptance rate; hence, the recognition rate decreased. It was at these thresholds (89.5%, 82.83%, 73.37%, 93%) for the same numbers of users, respectively.

The difference between the user group with 10 users and the user group with 40 users was 19.63%, the difference between the user group with 10 users and the user group with 30 users was 10.17%, and the difference between the 10-user group and 20-user group was 3.5%. Comparing the result of the 20-user group to the 30-user group and 40-user group, the difference was 6.67% and 16.3%, respectively, and the difference between the 30-user group and 40-user group was 9.46%. It is concluded that as the number of users increases, the user recognition rate decreases, as shown in Figure 2.

The user recognition rate was improved using the genetic algorithm, and the following rates were obtained (94%, 90.75%, 83.83%, 74.875%) with the same number of users and with the same thresholds previously used. The improvement ratio between the use of the genetic algorithm and non-use of it was (1%, 1.25%, 1%, and 1.5%) with user groups numbered 10, 20, 30, and 40, respectively.

A false acceptance rate of zero is maintained. The effect of emotions and feelings affecting the voiceprint was reduced by applying the genetic algorithm, and the security and reliability of the system were maintained. The application of the genetic algorithm did not allow persons denied entry into a building to enter, and the practical results proved that the method was working. Reducing the threshold without applying the genetic algorithm increases the false rejection rate for the number of user groups (10,20,30,40). The difference between the 40-user group and the user groups (10, 20, 30) was (9.46%, 16.13%, and 19.63%), respectively; the difference between the 30-user group and user groups (10, 20) was (6.67%, 10.17%), respectively; and the difference between 20-user group and 10-user group was 3.5%.

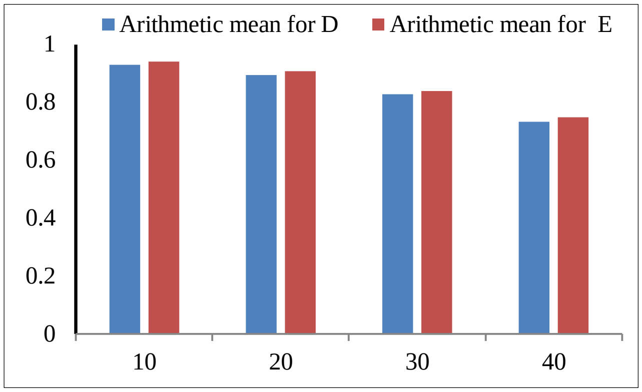

Fig. 2 User recognition rate using Euclidean distance function and genetic algorithm

By applying the genetic algorithm, the false rejection rate was reduced to the following values for the same numbers of users and at the same thresholds (6%, 9.25%,16.16%, and 25.125%). As a result, the difference between the 40-user group and user groups numbered 10, 20, and 30 was (19.125%, 15.875%, and 8.965%), respectively.

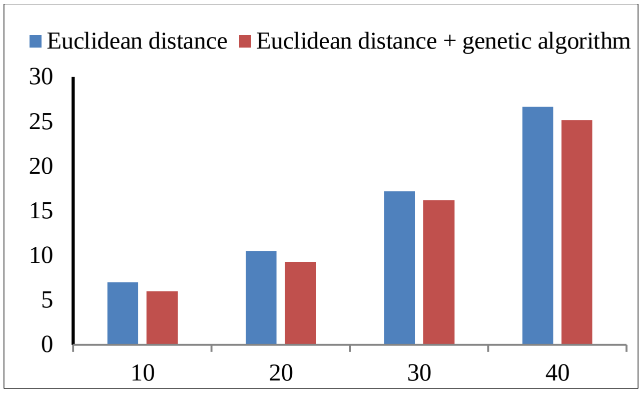

Fig. 3 False user rejection rate by using Euclidean distance function and genetic algorithm

The difference between the 30-user group and both the 20-user and 10-user groups was (10.16% and 6.91%), respectively, and the difference between the 20-user group and the 10-user group was (3.25%). Therefore, as the number of users increased and the threshold decreased, the rate of false rejection increased. Applying the genetic algorithm reduced this rate at the same thresholds and the same number of users, as shown in Figure 3.

The average time for voice analysis and voiceprint extraction was (3.183, 3.174, 3.171, and 3.169 sec.) for the user groups (10, 20, 30, 40), respectively, and at thresholds achieving the previously mentioned false acceptance rate of zero. The average test time was (0.00807, 0.00808, 0.0082, 0.0258 sec.) for the same number of users applying the Euclidean distance function.

The average test time was (0.00615, 0.023711, 0.020747, 0.022438) by applying the Euclidean distance function and the genetic algorithm for the same numbers of users, so the average arithmetic time for voice analysis and decision-making was (3.191, 3.182, 3.179, 3.195 sec.) by applying distance function, and was (3.18932, 3.19655, 3.192193, 3.192305) by applying the Euclidean distance function and the genetic algorithm. The process of testing and decision-making was accelerated by using the Euclidean distance function or by applying the Euclidean distance function, and the genetic algorithm was accelerated in a small, user-friendly time.

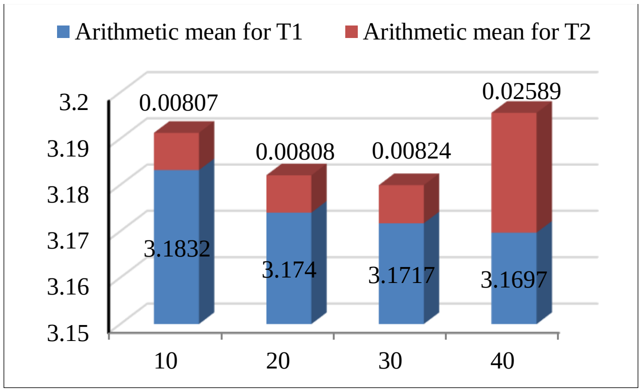

Fig. 4 Arithmetic mean of voiceprint analysis and testing time

5.2.1. Comparison of Current Studies with Previous Studies

This section compares the results obtained in this study with results from some previous studies using different voice analysis algorithms and test algorithms. The following table shows user recognition rate results for the sample groups and the methods applied:

Table 6. Comparison of current studies with previous studies

User Numbers	Results in the Current Study	Results in the Current Study Using GA (%)	MFCC+ANN Using BPNN[1]	MFCC+MNN on (QS-Dataset) [11]	MFCC+CNN on (QS-Dataset) [11]	MFCC+MNN on (Audio MNIST_meta) [11]	MFCC+CNN on (Audio MNIST_meta) [11]
10	93	94	92	88	89	76	79
20	89.5	90.75	82	85	96	66	81
30	82.83	83.83	76	81	96	56	82
40	73.37	74.875	72	-	-	-	-

Where,
ANN : Artificial Neural Network,
MNN : Multilayer Neural Network, and
CNN : Convolution Neural Network.

To check the identity of the speakers, the two researchers used a text-based method (predetermined words or phrases) [11], the same as that used in this study. To obtain voice features, they used the MFCC algorithm with fixed coefficients. They used 16 constant coefficients of this algorithm, which is the same as that used in this study. They used the ANN algorithm in two stages: a training stage and a testing stage.

The study was applied to different groups of speakers whose voices were recorded. As shown in the table above, the results of this research outnumbered Wali and Hatture's study. For the user numbers mentioned (10, 20, 30, 40), the recognition rate was 1%, 7.5%, 6.8%, and 1.37% better by using the Euclidean distance function and 2%, 8.75%, 7.8%, and 2.8% better by using the Euclidean distance and genetic function.

The two researchers also designed a text-based speaker recognition system. The voice features were extracted by using the MFCC algorithm. The study was applied to two databases, QS-Dataset and audioMNIST-met. The study was tested using two types of neural networks: Multilayer Neural Network (MNN) and Convolutional Neural Network (CCN). Results were shown in the previous table for the user groups (10, 20, 30) but did not include the 40-user group.

The results reached in this research were better than those reached using MNN and the application of the database QS-Dataset for all groups with the same number of users; the recognition rate was better (5%, 4.5%, 1.8%) with the Euclidean distance function and (6%, 5.75%, 2.8%) with the Euclidean distance and genetic function.

By applying CNN to QS-Dataset, the recognition rate in the 10-user group in the current study was 4% better with the Euclidean distance function and 5% better with the Euclidean distance and genetic function, but with 20-user and 30-user groups, the results were better in the reference study; the recognition rate was 6.5% and 13.17% better than the current study recognition rate with the Euclidean distance function and was 5.25%, 12.17% better with the Euclidean distance and genetic function.

By comparison of the results of the current research with the research of the audio MNIST_meta database, the current research results were better. For all groups using either MNN or CNN, identification of the speaker with the current study was 17%, 23.5%, and 26.83% better by applying the Euclidean distance function than with MNN, and it was 18%, 24.75%, 27.83% better by applying Euclidean distance and genetic function than with MNN. Also, the recognition rate for this study was 14%, 8.5%, and 0.83% better when applying the Euclidean distance function than CNN in the reference study, and it was 15%, 11.75%, and 1.8% better by using Euclidean distance and genetic function.

6. Conclusion

Applying the experimental statistical equation (dividing the mean by the standard deviation of the coefficient matrix) resulted in a rapid voice print because it shortened a large amount of data. Therefore, the decision is made quickly so the system is acceptable to the user and does not bore him.

- The problem of the unequal number of frames has been overcome, as the number of frames changes from one recording to another because it is affected by the noise in each recording and also by the speed of the speaker, and thus the number of frames that are deleted changes from one recording to another and the number of frames that are kept changes.
- By applying the statistical experimental equation, all the data that resulted from the voice analysis were considered, and no part of the data was ignored.

Acknowledgement

This work is supported by the University of Idlib, Syria, and Istanbul University, Turkey.

References

[1] S.S. Wali, S.M. Hatture, and S. Nandya, "MFCC Based Text-Dependent Speaker Identification Using BPNN," *International Journal of Signal Processing Systems*, vol. 3, no. 1, pp. 30-34, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[2] Mehmet Berkehan Akcay, and Kaya Oguz, "Speech Emotion Recognition: Emotional Models, Databases, Features, Preprocessing Methods, Supporting Modalities, and Classifiers," *Speech Communication*, vol. 116, pp. 56-76, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[3] Qiang Zhu et al., "Whispered Speech Conversion Based on the Inversion of Mel Frequency Cepstral Coefficient Features," *Algorithms*, vol. 15, no. 2, pp. 1-12, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Rajeshwari G. Dandage, and P.R. Badadapure, "Infant's Cry Detection Using Linear Frequency Cepstrum Coefficients," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 6, no. 7, pp. 5377- 5383, 2017. [CrossRef] [Publisher Link]

[5] W.S. Mada Sanjaya, Dyah Anggraeni, and Ikhsan Purnama Santika, "Speech Recognition Using Linear Predictive Coding (LPC) and Adaptive Neuro-Fuzzy (ANFIS) to Control 5 DoF Arm Robot," *Journal of Physics: Conference Series*, vol. 1090, pp. 1-10, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[6] Rashmi Kethireddy, Sudarsana Reddy Kadiri, and Suryakanth V. Gangashetty, "Exploration of Temporal Dynamics of Frequency Domain Linear Prediction Cepstral Coefficients for Dialect Classification," *Applied Acoustics*, vol. 188, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] Kranthi Kumar Lella, and Alphonse Pja, "Automatic Diagnosis of COVID-19 Disease Using Deep Convolutional Neural Network with Multi-Feature Channel from Respiratory Voice Data: Cough, Voice, and Breath," *Alexandria Engineering Journal*, vol. 61, no. 2, pp. 1319-1334, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] P.K. Nayana, Dominic Mathew, and Abraham Thomas, "Comparison of Text Independent Speaker Identification Systems Using GMM and i-Vector Methods," *Procedia Computer Science*, vol. 115, pp. 47-54, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[9] Harisudha Kuresan, and Dhanalakshmi Samiappan, "Genetic Algorithm and Principal Components Analysis in Speech-Based Parkinson's Early Diagnosis Studies," *International Journal of Nonlinear Analysis and Applications*, vol. 13, no. 1, pp. 591-602, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] Ismail Shahin, Ali Bou Nassif, and Noor Hindawi, "Speaker Identifcation in Stressful Talking Environments Based on Convolutional Neural Network," *International Journal of Speech Technology*, vol. 24, pp. 1055-1066, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[11] Qasim Sadiq Mahmood, and Yusra Faisal Al-Irahyim, "Text-Dependent Speaker Identification System Based on Deep Learning," *Journal of Education and Science*, vol. 30, no. 4, pp. 141-160, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[12] M. Subba Rao, K. Umamaheswari, and P. Venkata Jagadeesh, "Support Vector Machine Based Automatic Speaker Recognition System," *The International Journal of Analytical and Experimental Modal Analysis*, vol. 12, no. 3, pp. 1041-1049, 2020. [CrossRef] [Publisher Link]

[13] Yinchun Chen, "A Hidden Markov Optimization Model for Processing and Recognition of English Speech Feature Signals," *Journal of Intelligent Systems*, vol. 31, no. 1, pp. 716-725, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] Tsung-Han Tsai, Ping-Cheng Hao, and Chiao-Li Wang, "Self-Defined Text-Dependent Wake-Up-Words Speaker Recognition System," *IEEE Access*, vol. 9, pp. 138668- 138676, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[15] Rusydi Umar et al., "Identification of Speaker Recognition for Audio Forensic Using K-Nearest Neighbor," *International Journal of Scientific & Technology Research*, vol. 8, no. 11, pp. 3846- 3850, 2019. [Google Scholar] [Publisher Link]

[16] Anett Antony, and R. Gopikakumari, "Speaker Identification Based on Combination of MFCC and UMRT Based Features," *Procedia Computer Science*, vol. 143, pp. 250-257, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[17] Soufiane Hourri, Nikola S. Nikolov, and Jamal Kharroubi, "A Deep Learning Approach to Integrate Convolutional Neural Networks in Speaker Recognition," *International Journal of Speech Technology*, vol. 23, pp. 615-623, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[18] Feng Ye, and Jun Yang, "A Deep Neural Network Model for Speaker Identification," *Applied Sciences*, vol. 11, no. 8, pp. 1-18, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[19] Samia Abd El-Moneim et al., "Text-Independent Speaker Recognition Using LSTM-RNN and Speech Enhancement," *Multimedia Tools and Applications*, vol. 79, pp. 24013-24028, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[20] Zhanghao Wu et al., "Data Augmentation Using Variational Autoencoder for Embedding-Based Speaker Verification," *Proceedings of the Interspeech 2019*, pp. 1163-1167, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[21] Yuanjun Zhao, Roberto Togneri, and Victor Sreeram, "Multitask Learning-Based Spoofing-Robust Automatic Speaker Verification System," *Circuits, Systems, and Signal Processing*, vol. 41, pp. 4068-4089, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[22] K. Sreenivasa Rao, and K.E. Manjunath, *Speech Recognition Using Articulatory and Excitation Source Features*, Springer Briefs in Speech Technology, pp. 85-92, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[23] Rada A., Alhalabeia O., and Mansor A., "*Voice Recognition Using Neural Networks*," Thesis, Faculty of Electrical and Electronic Engineering, Aleppo University, Syria, 1999.

[24] Md. Afzal Hossan, Sheeraz Memon, and Mark A. Gregory, "A Novel Approach for MFCC Feature Extraction," *2010 4th International Conference on Signal Processing and Communication Systems*, Australia, pp. 1-5, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[25] Jane J. Stephan, "Speaker Identification Using Evolutionary Algorithm," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 13, no. 9, pp. 717-721, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[26] Sofia Kanwal, and Sohail Asghar, "Speech Emotion Recognition Using Clustering Based GA-Optimized Set," *IEEE Access*, vol. 9, pp. 125830-125842, 2021. [CrossRef] [Google Scholar] [Publisher Link]