Text-dependent Speaker Recognition System Based on Speaking Frequency Characteristics

Khoa N. Van[1], Tri P. Minh[1], Thang N. Son[1](✉), Minh H. Ly[1](✉), Tin T. Dang[1], and Anh Dinh[2]

[1] Faculty of Electrical and Electronics Engineering,
Ho Chi Minh City University of Technology, Vietnam National University,
Ho Chi Minh City, Vietnam
{1413647,1411312}@hcmut.edu.vn

[2] Department of Electrical and Computer Engineering, University of Saskatchewan,
Saskatoon, Canada

Abstract. Voice recognition is one of the various applications of Digital Signal Processing and has many important real-world impacts. The topic has been investigated for quite a long time and is usually divided into two major divisions which are speaker recognition and speech recognition. Speaker recognition identifies the person who is speaking based on characteristics of the vocal utterance. On the other hand, speech recognition focuses on determining the content of the spoken message. In this project, we designed and implemented a speaker recognition system that identifies different users based on their previously stored voice samples. The samples were gathered and its features were extracted using the Mel-frequency Cepstrum Coefficient feature extraction method. These coefficients, which characterize its corresponding voice, would be stored in a database for the purpose of later comparison with future audio inputs to identify an unknown speaker. The module is currently designed to be used as a standalone device. In the future, the module is equipped with the Internet of Things (IoT) for various security systems based on human biometrics.

Keywords: Voice recognition · Mel-frequency cepstrum coefficient
Gaussian mixture model · Discrete fourier transform

1 Introduction

Speech is recognized based on evidence that exists at various levels of the speech knowledge, ranging from acoustic phonetics to syntax and semantics [1]. While the field of speech recognition mainly focuses on converting speech into a sequence of words using computer [2] to extract the underlying linguistic message in an utterance, the field of speaker recognition is concerned with the identification of a person from the characteristics of his/her voice [3]. The speaker

T. K. Dang et al. (Eds.): FDSE 2018, LNCS 11251, pp. 214–227, 2018.
https://doi.org/10.1007/978-3-030-03192-3_16

recognition task falls into two categories which are text-dependent recognition and text-independent recognition. Each speaker recognition system is subdivided into two steps which are identification step and verification step [4]. In identification step, the goal is to record the speaker's voice and then extract a number of features from this voice to form a speaker's voice model. On the other hand, in the verification step, a speech sample is passed through all the previously created speaker's voice models for the purpose of comparison. If the lexicon inputs in two phases are the same, the system is called text-dependent speaker recognition as opposed to text-independent speaker recognition in which no conditional constraint is put on the input lexicon [5]. Successes in the tasks depend on extracting and modeling the speaker-dependent characteristics of speech signal which can effectively distinguish one speaker from another.

In this paper, a new speaker model combines the Mel-frequency Cepstrum Coefficient (MFCC) feature extraction methodology and Gaussian mixture speaker model will be introduced for text-dependent speaker recognition. The MFCC features are the most commonly used features in speaker recognition [6]. On the other hand, because of the accuracy in extracting and representing some general speaker-dependent spectral shapes and the capability to model arbitrary densities, the Gaussian mixture models (GMM) have been used largely for modeling speaker identity. For example, in [7], Leonardo Gongora has developed a system for extracting speech features, based on the MFCC and in [8], Reynol has used GMM to build a robust system for text-dependent speaker identification and achieved highly accurate result. Moreover, several different methods for voice recognition such as Unimodal Gaussian model [9], vector quantization (VQ) codebook [10] have been compared with GMM, and GMM demonstrates its outperformance for voice recognition (VR) tasks over other methods [8].

In this paper, we present an embedded system that uses the Elechouse V3 module integrated with the two algorithms mentioned above for the purpose of speaker recognition. The paper is organized into five sections. After the introduction, Sect. 2 depicts the theoretical aspect of the method and mechanism to implement an effective algorithm to identify a speaker. Section 3 proposes the structure and design of the module. Section 4 shows the experimental results, the corresponding accuracy as well as the comparison with other systems. Finally, Sect. 5 presents some observations and conclusions of the work and then proposes further improvements.

2 Speaker Recognition System

Voice is arguably the most basic form of human communication [11]. Human voice or speech is an information-rich signal that conveys a wide range of information such as the content of the speech, the feelings of the speaker, the tone of the speech, etc. [12]. The goal of the Speaker Recognition (SR) is to extract, describe and identify the speaker based on the voice characteristics. There are many systems which can facilitate the process of speaker identification. Generally, these systems contain two phases which are identification phase and verification phase. In the identification phase, each speaker's voice is collected and used to form that speaker's corresponding model. The set of all speakers' voice models is called the speaker database (Fig. 1).

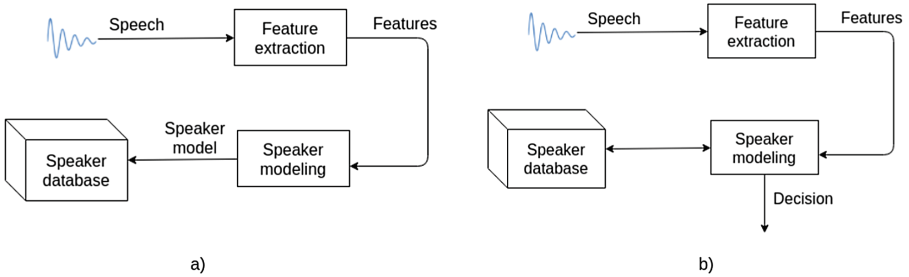


Fig. 1. Speaker recognition system: (a) The identification phase; (b) The verification phase.

Finally, in verification phase, the unknown speaker's voice data is put into the system and compared with all the models in the speaker database.

The two phases have two steps in common. The first step is to collect the voice, which can be collected through the microphone and digitalized to become discrete signals or digital signals. The second step is extraction, aiming to reduce the size of the data but still ensures sufficient information to identify the speaker. In the last step of the identification phase, the speaker's data is modeled using the GMM method and then stored in the database. Finally, in the last step of the verification phase, the extracted data is compared with all the models in the database to output the prediction about the speaker based on the likelihood. We present the above ideas of MFCC and GMM algorithms with more insights in the following sections.

2.1 Mel-Frequency Cepstrum Coefficient

Voice Signal Encoding. The simplest way of voice signal encoding is to encode the voice signal by approximating sound waves with a sequence of bytes representing the corresponding oscillation amplitude at equally spaced time intervals which are sufficiently small to maintain the sound information. This time interval unit is called the sampling rate. Figure 2(a) describes the different sampling rates that can affect the amount of information being captured which is used later in the verification phase. The value at each sampling is expressed in a specified, discrete value range called bit depth. The larger the bit depth range, the higher the identity between the sampled information and the real information (Fig. 2(b)).

Voice Feature Extraction. Speech signal covers a wide variety of information about the speaker which includes 'high-level' information such as language,

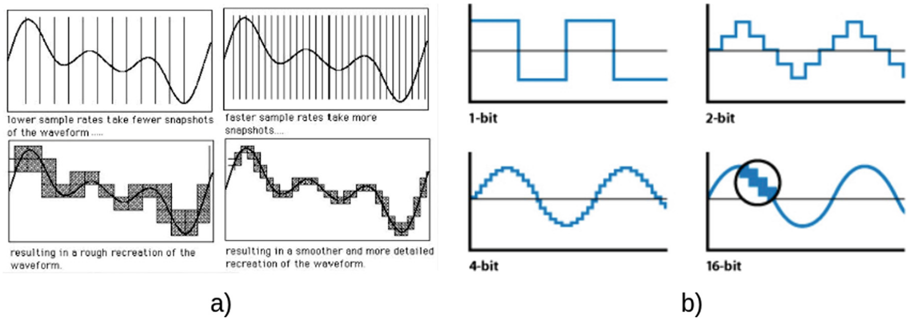

lower sample rates take fewer snapshots of the waveform .....
faster sample rates take more snapshots....
resulting in a rough recreation of the waveform.
resulting in a smoother and more detailed recreation of the waveform.
1-bit
2-bit
4-bit
16-bit
a)
b)

Fig. 2. Voice signal encoding: (a) The different sample rates [13] (b) The different bit depths.

context, spoken language, mood, etc. High level features contain more specific speaker-dependent information, but their extraction is very complicated [13]. Instead, low-level information such as pitch, intensity, frequency, band, audio spectrum, etc. can be easily extracted and are found to be very effective for the implementation of automatic SR systems [13]. The MFCC algorithm proposes an efficient way to extract sufficient information to distinguish one person from another.

Mel-Frequency Cepstrum Coefficients (MFCC) Feature Extraction. MFCC is based on the evidence that information carried by the low-frequency components is more phonetically important than the high-frequency sounds [14]. Figure 3 describes the step-by-step algorithm to extract the MFCC features.

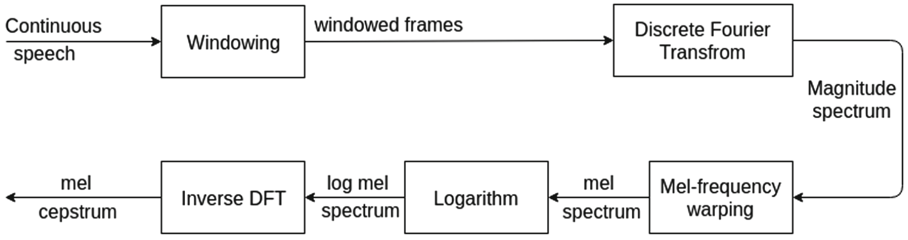


Fig. 3. MFCC feature extraction algorithm.

Windowing. In the first step, the original voice signal is divided into consecutive frames where these frames are sampled at equally spaced time intervals. Each frame's features will then be extracted using the MFCC model. In reality, the voice varies slowly, therefore if the analysis is implemented on sufficiently short time (20–30 ms) then the voice's characteristic will be stable. Features extracted at these time intervals characterize the speaker's voice. This process is called the short-term analysis.

Discrete Fourier Transform. Each frame obtained after the first step will be passed through a Discrete Fourier Transform (DFT). After performing DFT, another voice's characteristic called cepstrum is obtained. Due to the unique complexity of one person's voice, each voice has its own fingerprint qualities, i.e., no two people's voice patterns are exactly similar [16]. The method of Fourier Transform analysis maps the frequency fingerprint of a person's voice, in order to distinguish one voice from another [16].

Mel-Frequency Filtering. The signal information in terms of frequency and intensity is obtained after the DFT transform. A frequency scale which is called the mel-frequency is used to measure the perception of human ear. One of the most widely used formulas to convert from Hz to mel is from Lindsay and Loman [17]:

$$m = 2410 \log_{10}(1.6 \times 10^{-3} f + 1) \tag{1}$$

where f is the value of the frequency in Hz and m is the value of the frequency in mel.

Logarithm and Inverse Discrete Fourier Transform. Voice signal can be represented by two components which are fast-changing E and slow-changing H components [18]. It is possible to express the correlation of these two fast and slow information as follows:

$$|S(x)| = |E(x) * H(x)| \tag{2}$$

Where $E(x)$ is the high-frequency (fast-changing) component, $H(x)$ is the low-frequency (slow-changing) component and $S(x)$ is the original signal. The above expression can be translated to addition using logarithm:

$$\log_{10}(|S(x)|) = \log_{10}(|E(x)|) + \log_{10}(|H(x)|) \tag{3}$$

After this operation, the inverse Discrete Fourier Transform (IDFT) is performed on $\log_{10}(|S(x)|)$. As a result of this transform, one can separate two regions with high and low frequencies. The frequency region needed is the low frequency. Figure 4 depicts the idea.

2.2 Gaussian Mixture Model (GMM) and Speaker Recognition

Gaussian Distribution Model and Gaussian Mixture Model. The Gaussian mixture model is defined as the weighted sum of M components:

$$p(\boldsymbol{x}|\lambda) = \sum_{i=1}^{M} b_i p_i(\boldsymbol{x}) \tag{4}$$

Where b_i is the mixture weight, $p_i(\boldsymbol{x})$ is the probability density of the i^{th} component with vector $\boldsymbol{x}$ whose length is N and M is the total number of components.

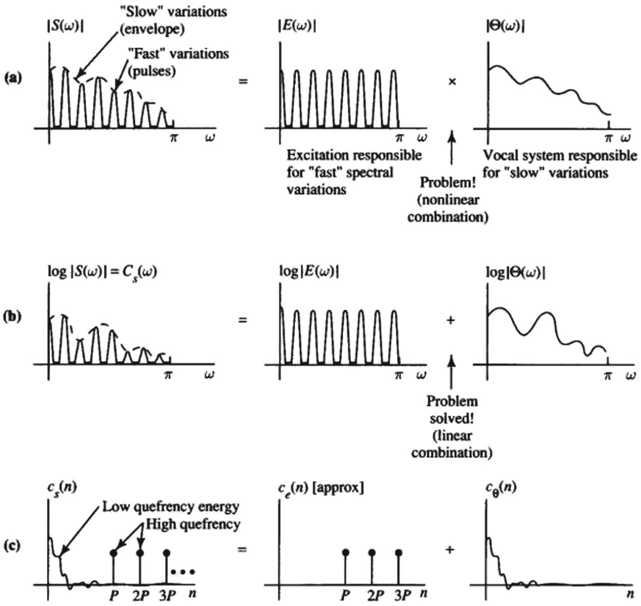

(a)
$|S(\omega)|$
"Slow" variations (envelope)
"Fast" variations (pulses)
=
$|E(\omega)|$
×
$|\Theta(\omega)|$
Excitation responsible for "fast" spectral variations
Problem! (nonlinear combination)
Vocal system responsible for "slow" variations
(b)
$\log|S(\omega)| = C_s(\omega)$
$\log|E(\omega)|$
+
$\log|\Theta(\omega)|$
Problem solved! (linear combination)
(c)
$c_s(n)$
Low quefrency energy
High quefrency
$c_e(n)$ [approx]
$c_\theta(n)$
P 2P 3P n
π ω

Fig. 4. Inverse Discrete Fourier Transform and separating the low and high frequencies components [15].

The sum of b_i equals 1. In other words, $\sum_{i=1}^{M} b_i = 1$. Each component density $p_i, i = 1, \ldots, M$ is a multivariate Gaussian normal distribution as follows:

$$f_{\boldsymbol{x}}(x_1, x_2, \ldots, x_N) = p_i(\boldsymbol{x}) = \frac{1}{\sqrt{|\Sigma_i|(2\pi)^N}} \exp\left(\frac{-(\boldsymbol{x}-\boldsymbol{\mu_i})^T(\Sigma_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu_i}))}{2}\right) \tag{5}$$

Where $\boldsymbol{x}$ is the random vector and $\boldsymbol{\mu_i}$ is the expected vector (mean vector) both with length N. Σ_i is the covariance matrix of size N x N. The right hand side product:

$$(\boldsymbol{x}-\boldsymbol{\mu_i})^T(\Sigma_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu_i})) \tag{6}$$

would produce a 1×1 matrix, i.e., a real number value.
The Gaussian Mixture Model is characterized by three parameters [8]:

$$\lambda = \{b_i, \mu_i, \Sigma_i\} \tag{7}$$

where

$$i = 1, 2, \ldots, M$$

For speaker recognition, each speaker is represented by a GMM and is referred to by his/her model λ.

Speaker Modeling with Gaussian Mixture Model. The use of GMM allows the representation of a large number of different models corresponding to different speakers. Each speaker's model is formed based on its corresponding MFCC's vector which is extracted in the feature extraction phase. The most commonly used method for finding the coefficients of the Gaussian model is the Maximum Likelihood Estimation method, which is the method of finding one or more parameters for a given statistic which maximizes the known likelihood distribution [19]. Concretely, for a sequence of T training vectors $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_T\}$, the likelihood can be written as:

$$p(X|\lambda) = \prod_{t=1}^{T} p(\boldsymbol{x}_t|\lambda) \tag{8}$$

Speaker Recognition. Once the model for each corresponding speaker is built, the system can be used to identify a speaker with new input data. The data should be preprocessed, feature extracted and compared with all the built models stored in the database. Assume that there is a set of S speakers {1, 2,. . . , S} with S corresponding GMM models $\{\lambda_1, \lambda_2, ..\lambda_S\}$. The goal is to find the model which outputs the highest probability with a new specific voice as input. This is the model that matches the unknown voice with highest confidence.

$$\hat{S} = \underset{1 \leq k \leq S}{\operatorname{argmax}} \Pr(\lambda_k|X) \tag{9}$$

3 Embedded System Design

3.1 Speaker Recognition Module and Microcontroller

Speaker Recognition (VR) Module Elechouse V3. In this VR embedded system, the Elechouse V3 module, currently one of the most compact voice control modules, is used. The V3 board can support up to 80 voice samples each with a duration of 1500 milliseconds. The module would compare a speaker's voice with a set of recorded voices. The module can obtain results with 99% recognition accuracy under ideal conditions [20]. The choice of microphone and the noise conditions can substantially affect the performance of the module.

Arduino Microcontrollers. The Arduino Nano and Arduino Uno R3 boards are used for the proposed embedded system. These boards are equipped with a set of digital and analog inputs and outputs where serial communications interfaces, including Universal Serial Bus (USB), are supported [21,22].

3.2 Speaker Recognition Embedded System Design

From the perspective of hardware design, the system design is divided into four units including the transmitting operation unit, the processing unit, the sampling unit, and the output unit. The system design schematic is as illustrated in Fig. 7 (Figs. 5 and 6).

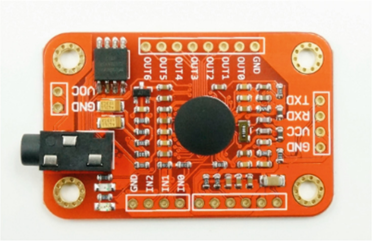
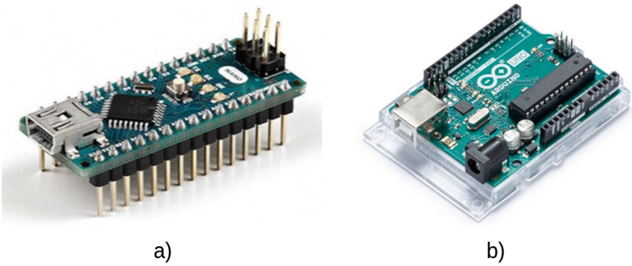


Fig. 5. Voice recognition module Elechouse V3 [20].


a)

b)

Fig. 6. (a): Arduino Nano microcontroller and (b): Arduino Uno R3 microcontroller [23].

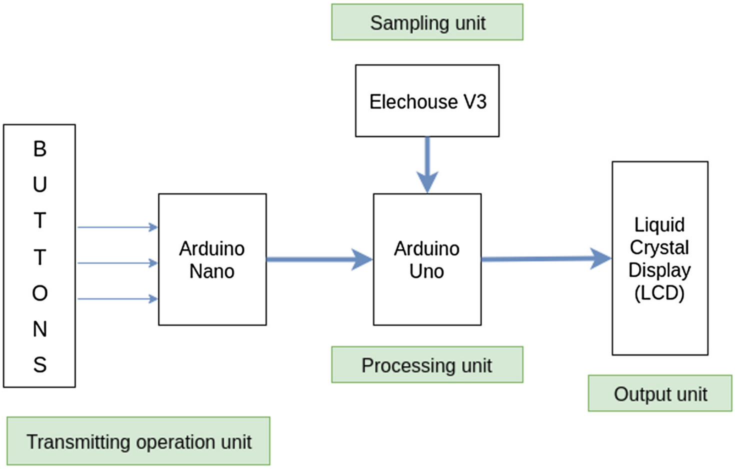


Fig. 7. Speaker recognition embedded system design.

Transmitting Operation Unit. There are three main operations which are speaker's voice sampling operation, speaker's sample loading operation and sample deleting operation. Each operation can be implemented by pushing the corresponding button on the Arduino Nano board.

Sampling Unit. Elechouse V3 is tasked to record every speaker's voice sample using a microphone. It is required that for each speaker, the sampling process repeats two times. In order to have good results, the environment noise should be reduced when the Elechouse module is recording the speaker's voice.

Processing Unit. The Arduino Uno R3 is used as the main controller, which receives commands from the Arduino Nano and data sent from the Elechouse V3. The controller would handle the received data and then display the output on the LCD (liquid crystal display) screen.

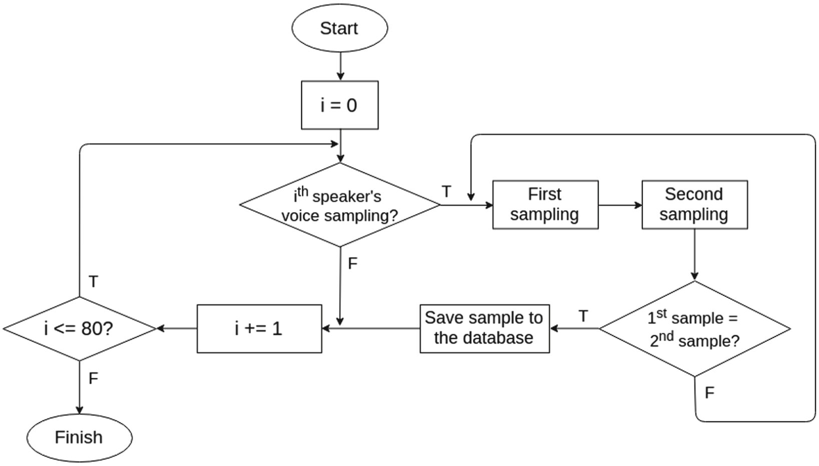


Fig. 8. Identification phase.

As mention above, the proposed algorithm is divided into two phases which are the identification phase and verification phase. Figure 8 describes the identification phase where the system records and samples the speaker's voice. The speakers' voices are sampled consecutively. For each speaker, the sampling process repeats two times. If the lexicon input at the first time is different from the second or these inputs are from different speakers, the system discards this sampling and start a new sampling. The speaker's voice sample is then stored in the Elechouse V3's memory unit.

In the second phase which is the verification phase, the aim is to recognize the speaker's voice. Firstly, all voice samples from the memory are loaded into the microcontroller. After that, the speaker whose voice needs to be recognized is recorded and sampled. This sample is then compared with all the voice samples stored in the database. If there exists a high similarity between two voice samples, the system would output the corresponding speaker's ID on the screen, otherwise it would return a null value. Figure 9 depicts this verification phase. The dataset size was limited to at most 80 individuals' voice sample because the Elechouse Module V3 can support up to 80 voice samples [20].

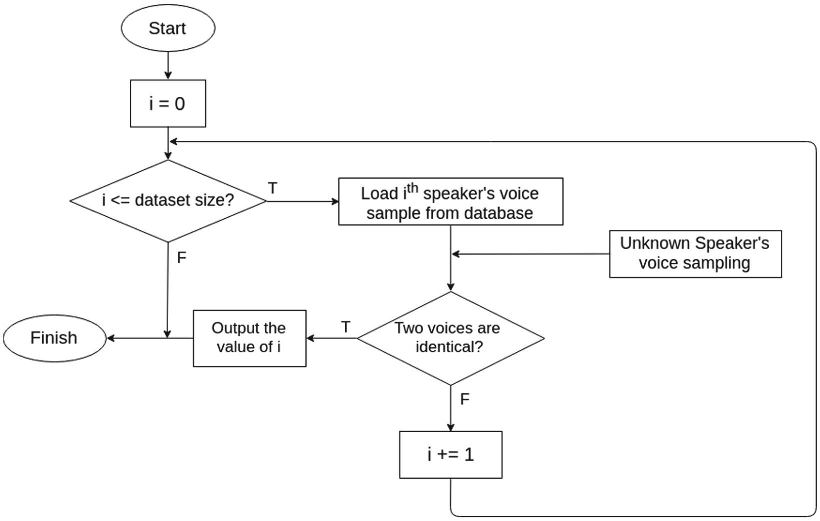

Start
i = 0
i <= dataset size?
T
F
Load i^th speaker's voice sample from database
Unknown Speaker's voice sampling
Two voices are identical?
T
F
Output the value of i
Finish
i += 1

Fig. 9. Verification phase.

4 Results

4.1 Integrated System

The system was designed and implemented as shown in Fig. 10. In addition to 2 Arduino microcontrollers and the Elechouse V3 module, one LCD screen, one microphone and one battery power supply are required.

4.2 Experimental Result

The system was tested on 10 speakers (5 men and 5 women with the ages between 15 and 60), all under noisy and close-to-ideal (CTI) environment circumstance. Each person is tested 50 times. Then we evaluated the received results by comparing it with the other current systems, results were taken from [8] as the reference.

Table 1 shows that the proposed system achieved a very good accuracy (approximately 90%) in close-to-ideal environmental circumstances (i.e. without noise). This highly accurate result can be substantially affected by noise,

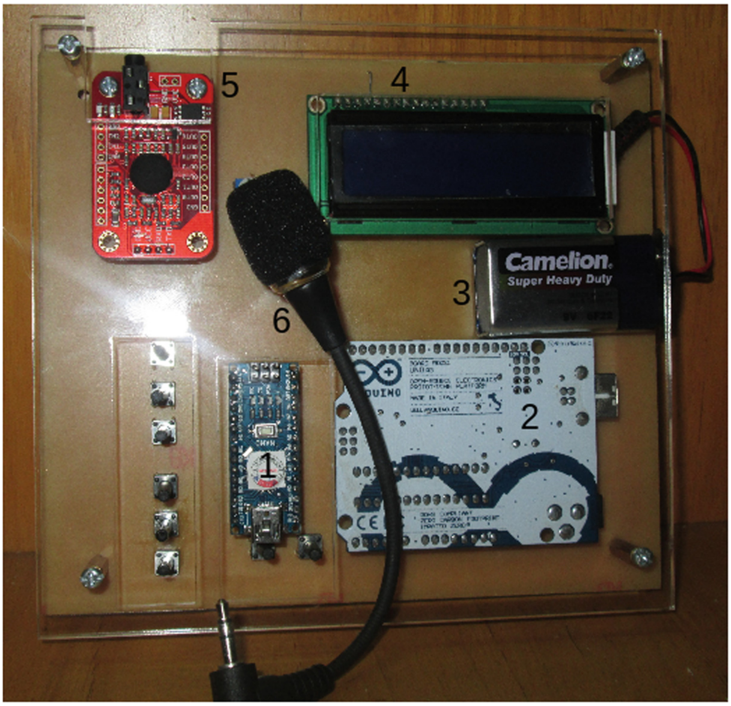


Fig. 10. The implemented embedded system with 1 - Arduino Nano; 2 - Arduino Uno R3; 3 - The power supply; 4 - The LCD screen; 5 - The Elechouse V3 module and 6 - The microphone.

Table 1. Embedded system test results

Speaker	Noisy environment		CLI environment	
	*Successes**	*Accuracy*	*Successes**	*Accuracy*
1	35	70%	45	90%
2	36	72%	44	88%
3	35	70%	45	90%
4	38	76%	45	90%
5	34	68%	45	90%
6	36	72%	44	88%
7	35	70%	46	92%
8	37	74%	44	88%
9	36	72%	45	90%
10	37	74%	45	90%

*Successes over 50 trials

which could decrease the accuracy to about 70%. However, in both noisy and close-to-ideal environment circumstance, the proposed system gave a small standard deviation value (σ) in the overall result compared to other methods, as shown in Table 2.

Table 2. Comparison to other SR methods, results taken from [8]

Methods	Accuracy (%)
GMM-nv	94.5 ± 1.8
VQ-100	92.9 ± 2.0
Our system under CTI environment circumstance†	89.6 ± 0.86
GMM-gv	89.5 ± 2.4
RBF	87.2 ± 2.6
TGMM	80.1 ± 3.1
Our system under noisy environment circumstance†	71.8 ± 1.85
GC	67.1 ± 3.7

†99% confidence interval

5 Conclusion

This paper has introduced and evaluated the design and operation of an embedded system using MFCC feature extraction method along with GMM model for robust text-dependent speaker recognition. The primary focus of this work is for real-world applications, such as for home security. The experimental results evaluated several aspects of using GMM models along with MFCC method for speaker recognition. Some observations and conclusions are:

- The proposed system achieved high accuracy compared to other contemporary methods. However, the overall result depends on the environment circumstance. In other words, this accuracy rate is affected by the noise condition
- To improve the accuracy of the system, state-of-the-art machine learning algorithms can be implemented to extract the voice's features along with GMM model instead of using MFCC feature extraction method.
- For the purpose of future improvement, the authors suggest programming the module to be able to take voice sample of the speaker from audio file through various media devices such as smartphone, tablet or laptop, using IoT technology. This would help the sampling step to be implemented even when the speaker is not present and help discard the necessity of using a microphone.

References

1. O'Shaughnessy, D., Deng, L., Li, H.: Speech information processing: theory and applications. Proc. IEEE **101**(5), 1034–1037 (2013)
2. Huang, X., Deng, L.: An overview of modern speech recognition. In: Handbook of Natural Language Processing, pp. 339–366. CRC Press, New York (2010)
3. Poddar, A., Sahidullah, M., Saha, G.: Speaker verification with short utterances: a review of challenges, trends and opportunities. IET Biom. **7**(2), 91–101 (2018)
4. Sara Rydin page. http://www.speech.kth.se/~rolf/gslt_papers/SaraRydin.pdf. Accessed 3 June 2018
5. Larcher, A., Lee, K.A., Ma, B.: Text-dependent speaker verification: classifiers, databases and RSR2015. Speech Commun. **60**, 56–77 (2017)
6. Yankayis, M.: Feature Extraction Mel Frequency Cepstral Coefficient (MFCC). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.701.6802&rep=rep1&type=pdf. Accessed 3 June 2018
7. Gongora, L., Ramos, O., Amaya, D.: Embedded mel frequency cepstral coefficient feature extraction system for speech processing. Int. Rev. Comput. Softw. **11**(3) (2016)
8. Reynolds, D.A., Rose, R.C.: Robust text-dependent speaker identification using Gaussian mixture speaker models. IEEE Trans. Speech Audio Process. **3**(1), 72–83 (1995)
9. Soong, F., Rosenberg, A., Rabiner, L., et al.: A vector quantization approach to speaker recognition. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Florida, vol. 1, pp. 387–390 (1985). https://doi.org/10.1109/ICASSP.1985.1168412
10. Gish, H., Karnofsky, K., Krasner, M., et al.: Investigation of text-independent speaker identification over telephone channels. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Florida, vol. 1, pp. 379–382 (1985). https://doi.org/10.1109/ICASSP.1985.1168410
11. Truax, B.: Voices in the soundscape: from cellphones to soundscape composition. In: Electrified Voices: Medial, Socio-Historical and Cultural Aspects of Voice Transfer, pp. 61–79. V&R Unipress, Gottingen (2013)
12. Johar, S.: Emotion, Affect and Personality in Speech: The Bias of Language and Paralanguage. Springer, New York (2016). https://doi.org/10.1007/978-3-319-28047-9
13. Animemusicvideos's Understanding Audio Homepage. https://www.animemusicvideos.org/guides/avtech/audio1.html. Accessed 3 June 2018
14. Nayana, P.K., Mathew, D., Thomas, A.: Comparison of text independent speaker identification systems using GMM and i-vector methods. Procedia Comput. Sci. **115**, 47–54 (2017)
15. Deller, J.R., Hansen, J.H.L., Proakis, J.G.: Discrete-Time Processing of Speech Signals. IEEE Press, New Jersey (2000)
16. Kohanski, M., Lipski, A.M., Tannir, J., Yeung, T.: Development of a voice recognition program. https://www.seas.upenn.edu/~belab/LabProjects/2001/be310s01t2.doc. Accessed 3 June 2018
17. Lindsay, P.H., Norman, D.A.: Human Information Processing: An Introduction to Psychology, 2nd edn. Academic Press, New York (1977)
18. Feng, L.: Speaker Recognition, Master Thesis at Technical University of Denmark, Kongens Lyngby (2014). http://www2.imm.dtu.dk/pubdb/views/edoc_download.php/3319/pdf/imm3319.pdf. Accessed 3 June 2018

19. Wikipedia's Maximum likelihood estimation Homepage. https://en.wikipedia.org/wiki/Maximum_likelihood_estimation. Accessed 3 June 2018
20. Elechouse Voice Recognition Module V3 datasheet. https://www.elechouse.com/elechouse/images/product/VR3/VR3_manual.pdf. Accessed 3 June 2018
21. Arduino Nano V2.3 datasheet. https://www.arduino.cc/en/uploads/Main/ArduinoNanoManual23.pdf. Accessed 3 June 2018
22. Arduino Uno datasheet. https://www.farnell.com/datasheets/1682209.pdf. Accessed 3 June 2018
23. Arduino store. https://store.arduino.cc/. Accessed 3 June 2018