IET Signal Processing

Extracting an Identity-Vector for Speaker Verification Using the Time-Frequency and MFCC Methods in the Presence of Hardware Impairments and Non-Gaussian Noise

Journal:	*IET Signal Processing*
Manuscript ID	SPR-2022-02-0038
Wiley - Manuscript type:	Original Research
Date Submitted by the Author:	17-Feb-2022
Complete List of Authors:	Tavakkoli Kalatehno, Alireza; Shahrood University of Technology, Electrical and Computer Engineering Marvi, Hossein; Shahrood University of Technology
Keywords:	speaker recognition, time-frequency analysis, noise, distortion, impulse noise, discrete wavelet transforms
Abstract:	The advances of speaker verification and identification systems alongside the efforts in improving the performance of such systems in real environments severely necessitate the analysis and usage of such systems in real and applicable situations. This paper considers two challenges to analyze the real conditions of speaker verification systems. The first challenge relates to non-Gaussian noise, which is rather important considering the interferences of the current world, and the second relates to hardware impairment, which comes out of the impossible nature of finding the perfect system considering today's cheap software. Then, we extract the features in the context of the identity-vector (I-vector) using the time-frequency analysis and Mel frequency cepstral coefficients (MFCC) methods. Finally, we use the TIMIT data base to compare the performance of the proposed system with others using the equal error rate (EER) point criteria.

Extracting an Identity-Vector for Speaker Verification Using the Time-Frequency and MFCC Methods in the Presence of Hardware Impairments and Non-Gaussian Noise

Alireza Tavakkoli Kalatehno[a,*] and Hossein Marvi[b]

[a,b]Shahrood University of Technology, School of Electrical Engineering, Shahrood, Iran.

* Corresponding author. E-mail address: alirezatavakkoli@shaharoodut.ac.ir

Abstract: The advances of speaker verification and identification systems alongside the efforts in improving the performance of such systems in real environments severely necessitate the analysis and usage of such systems in real and applicable situations. This paper considers two challenges to analyze the real conditions of speaker verification systems. The first challenge relates to non-Gaussian noise, which is rather important considering the interferences of the current world, and the second relates to hardware impairment, which comes out of the impossible nature of finding the perfect system considering today's cheap software. Then, we extract the features in the context of the identity-vector (I-vector) using the time-frequency analysis and Mel frequency cepstral coefficients (MFCC) methods. Finally, we use the TIMIT data base to compare the performance of the proposed system with others using the equal error rate (EER) point criteria.

Keywords: Speaker Verification, Non-Identity Information, Identity-Vector, Time-Frequency Analysis, Non-Gaussian Noise, Hardware Impairments;

1. Introduction

The speech signal is an ideal way of human communication that contains different information. It could have a different application based on its information types, such as speaker identification, speaker verification, speaker emotions, accent, or language. The goal of speaker identification systems is to extract all of this information based on the speaker's voice to identify their identity. Speaker verification is one of the main objectives of speaker identification systems, which intended to verify or deny any claimant as to the system's intended person [1]. On the other hand, speaker verification systems are divided into text-independent or text-dependent groups. The text-independent systems have no limitations over the speech content and the user is not limited in expressing words or sentences. They require longer speeches for training and learning to reach a proper performance level. The text-dependent systems have the same content during the registration and test stages. The speaker expresses a limited set of words or sentences. In other words, this speech is the same as speech from the learning stage. These systems have a higher performance rate due to shorter speeches in the learning and testing stages [2].

The existence of hardware impairments is unavoidable in small, low-consumption, and inexpensive hardware. Speaker recognition have high security and rapid development rate in the modern world; therefore, the analysis of the hardware impairment challenge seems necessary in current systems. Phase noise, the quantization error, and non-linear amplifiers are some problems of low-consumption and cheap hardware [3-5].

The system noise is considered as Gaussian and accumulative in many statistical signal processing problems. This is while the accumulative system noise exists in a non-Gaussian and impulsive manner in real environmental conditions. Speech processing can contain non-Gaussian noises; therefore, their existence in the system could be beneficial in this manner [6].

1.1. Available Challenges and Project Goal

Speaker verification systems face different practical challenges such as hardware impairments and non-Gaussian noise. Therefore, researchers always present the following important questions regarding these systems:

1. What is the impact of hardware impairments on speaker authentication?
2. Considering the system interferences, Is the Gaussian noise model (or in other words, the central limit theorem) a practical and efficient hypothesis in these systems?
3. Time-Frequency analysis methods, such as Wavelet, have what benefits in feature extraction compared to other common methods?

This study aims to present a comprehensive answer to the above questions and challenges.

A review of previous studies shows the lack of joint or independent studies regarding the effects of hardware impairments and non-Gaussian noise on the performance of speaker authentication systems. As far as the authors know, this study is one of the first few works that analyze the joint challenges of hardware impairments and non-Gaussian noise in speaker authentication.

The most important innovations of this study are:

- Current microprocessors and processors are usually considered ideal; even though, hardware impairment is an inseparable part of electronic devices. Therefore, this study considers processors and microprocessors in their real state. As a result, hardware impairment is fully expressed in the system model for speaker authentication.
- Speech signals are usually affected by accumulative noise in the receiver system. Most of these signals are assumed as Gaussian signals, which is not a complete assumption. This study uses non-Gaussian accumulative noise because of different causes such as electromagnetic interferences, inter-user interferences, and others.
- Feature extraction is an important part of speaker authentication systems. This study uses the Time-Frequency analysis and MFCC methods such as Wavelet-MFCC, Gabor-MFCC, STFT-MFCC, Wigner-Ville-MFCC to extract speech features.

2. I-vector

The identity vector (I-vector) was first introduced for speaker recognition and moved into other fields after a while [7]. The I-vector is a vector with small dimensions extracted from a piece of speech including the speaker information, channel, and noise as a general concise representation of the important information in the speech signal. This method usually extracts different-length signals from a fixed-length vector based on joint factor analysis (JFA). In other words, it expresses the relationships between observable variables with some latent variables to reduce the super vector dimensions of the Gaussian mixture model (GMM) in the speaker authentication problem. Therefore, the usage idea of these vectors is the same as the super vector [8]. In reality, the voice segments are shown as vectors with fixed dimensions. Also, the Gaussian mixture model's super vector gains one speech segment from the mixture of Gaussian feature average vectors trained for each segment [9-10].

We extract the intended features from the speech signal to gain the I-vector of that speech signal. These features are usually a set of feature vectors. Then, we train the universal background model (UBM). Afterward, we extract some features from the set, calculate their Baum-Welch statistics, model their total variability space, and create the I-vector. The details of each stage are presented in the following. Figure 1 shows the block diagram of the proposed system.

2.1. Hardware Impairments

In reality and practice, the processes and microprocessors of current systems have different types of disturbances and impairments such as phase noise, quantization error, non-linear analog components in the amplifiers, and others. The usage of more expensive and precise hardware alongside more complex algorithms such as compensators can partly resolve these impairments but such equipment brings higher costs and power consumption. Even these equipment and methods are not capable of fully resolving all impairments, and systems will always have some impairment [4]. The most important reasons for the existence of hardware impairments in a system are [5]:

1. Incorrect modeling leading to an unsolvable error
2. Incorrect parameter estimations
3. A time-variable nature alongside random noise

Therefore, these hardware impairments are important for systems and must be considered when designing systems. This study assumes the usage of a proper compensator algorithm and focuses on the remaining joint hardware impairment effect left behind by the compensator. We assume the signal gets some distortion because of an accumulative Gaussian noise to model the residual effects of these hardware impairments. This model facilitates theoretical as well as practical empirical analysis. To be more precise, when there is a mixed effect left behind in the system by these hardware impairments, the speech signal received by the speaker verification system is as follows [5]:

$$r = s + \eta + \theta. \tag{2}$$

In above Equation, s signifies the speech signal and η shows the distortion caused by the unideal software in the speaker verification system, which is modeled as a Gaussian accumulative noise with an average of zero and variance of k^2 as follows:

$$\eta :\ CN\left(0, k^2\right), \tag{3}$$

In other words, k^2 determines the non-ideal hardware impairment value in the speaker verification system that is close to the square value of the error vector rate. This criterion measures the incompatibility between the real and intended signal in the speaker verification system. In some special cases, $k = 0$ shows the ideal hardware. Also, θ shows the non-Gaussian noise that is introduced in the next section.

2.2. Non-Gaussian Noise

Many statistical signal processing problems have sparse system parameters. For example, the channel impulse response of audio channels has a sparsity feature. On the other hand, audio channels are not stable over time and this channel impulse response changes over time. Also, many audio processing applications model system noise in an accumulative and Gaussian manner. This is while this accumulative system noise might be in a non-Gaussian or impulsive manner when it comes to real environments including calibration error and available disturbances. Speech processing applications are one of the environments for non-Gaussian noise. This study defines the speaker authentication system using Gaussian mixture distribution with the following probability density function [6]:

$$f(\theta) = \sum_{m=1}^{M} \frac{\varepsilon_m}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{\left(\theta - \mu_m\right)^2}{2\sigma_m^2}\right), \tag{4}$$

Which M shows the number of Gaussian components, and ε_m defines the weight of the m-th element with a limitation of $\sum_{m=1}^{M}\varepsilon_m = 1$. Here, the μ_m σ_m^2 parameters show the average and variance of the m-th element. Notably, this model can include different descriptions such as manmade impulsive noise for Telecommunication environment noises.

2.3. Feature Extraction

Each signal contains a wide array of diverse information, some of which only add to the computational load with no application in the system due to redundancy or uselessness. Therefore, feature extraction has two goals: first, it focuses on the related and useful information of the audio signal to increase inner-class similarities and reduce inter-class similarities; and second, it significantly reduces the data volume and calculations [11]. This leads to an important question, what are the benefits of time-frequency analysis methods compared to only time or frequency-based methods?

2.3.1. Time-Frequency Analysis Methods

Many engineering applications felt the need for a tool to analyze signals using the time and frequency variables. The goal of signal analysis is knowing the indicator behavior and features of a signal, and it is a favorite subject in the field of signal processing. Because when we draw the signal domain based on time, much useful information remains hidden in the frequency spectrum that we are unaware of. In other words, much useful signal information is hidden in the frequency domain in most cases. Therefore, the main goal of this analysis is extracting information and discovering the frequency domain content. This frequency changes over time so this act usually includes the observation of long-term behavior and short-term changes. The main tools for this operation are different transforms such as the Fourier transform. The Fourier transform is a powerful mathematical tool that breaks the signal into a mixed series of exponential functions with different frequencies (i.e., it takes the signal from time into frequency). The signal must be static with limited energy to undertake the Fourier transform. The Fourier transform uses a specific frequency to calculate the integral of the main function and exponential core multiplication over the entire timeframe. Therefore, the stronger integral values in any frequency show the stronger frequency spectrum elements of the signal at that specific point. We cannot determine the exact moment where a frequency element results in a non-zero integral whenever there is a frequency of f because the integral is calculated over the entire timeframe. Therefore, the Fourier transform is not appropriate for non-static signals. Also, this transform does not fit some applications because this transform requires temporal information regarding the past and future considering its analysis and synthesis relationships to extract frequency information. Also, it is impossible to determine specific periods that create a specific frequency range (i.e., it is impossible to say that the information from a specific frequency band belongs to which exact period) and on the other hand, it is impossible to say that any specific period made which spectrum information (i.e., we don't have the temporal and frequency information of the signal simultaneously).

Previous studies would ignore the non-static nature of some signals using some semi-static assumptions in the frequency field but the low accuracy and proficiency of these assumptions lead to the modern usage of time-frequency analysis methods. The time-frequency mapping shows an assumed time signal over a two-dimensional time-frequency coordinate system. This mixed mapping results from the time and frequency analysis that present a clearer picture of the location of signal spectrum time-frequency components. The time-

frequency analysis concept can show a single-dimensional signal using a bivariate joint function based on time and frequency variables. Most time-frequency mappings have a time-variable spectrum. Time-frequency surface values on the page show different information such as the existence time of different spectrums.
Different time-frequency methods have been recommended for usage in different fields such as non-static signal filtration, signal and system design, noise removal, image processing, compression, and others. The Wavelet method, which will be analyzed, later on, has the best performance compared to other methods such as Wigner-Ville, Gabor, and STFT. The feature extraction method of this paper is a combination of Wavelet and MFCC. The reason behind the usage of these two methods will be explained later on.

2.3.2. MFCC

Mel is a logarithm-based frequency unit. The human ear understands frequencies in a logarithmic manner; therefore, Mel is a rather useful unit for speech signal feature extraction. Cepstrum analysis is based on the Fourier transform, which is usually implemented in speech processing with two goals:

1. It compresses the samples.
2. It determines features with uncorrelated data.

This is why the MFCC feature extraction method is so popular nowadays.
This paper uses the MFCC method besides the Wavelet method. This method divides the signal into different frames with overlap to transform a non-static signal into a static one [13].

2.4. Universal Background Model (UBM)

The universal background model is the Gaussian mixture model. It aims to model the information distribution separate from each class. The Gaussian mixture model is trained with this aim using the feature extraction models of all audio signals. The Gaussian mixture model is a set of Gaussian functions with different weights used to model the density probability function. The D-dimensional x density probability function is as follows [14]:

$$f\left(x_n \mid \mu_g, \Sigma_g\right) = \sum_{g=1}^{N} \pi_g CN\left(x_n \mid \mu_g, \Sigma_g\right), \tag{5}$$

In which, π_g shows the g-th weight in the Gaussian mixture. $CN(.)$ shows the Gaussian distribution with an average of μ_g and covariance of Σ_g in a way that $\sum_{g=1}^{N} \pi_g = 1$. Notably, each density probability function component has a Gaussian distribution with D variables as follows:

$$CN\left(x_n \mid \mu_g, \Sigma_g\right) = \frac{1}{\left(2\pi\right)^{D/2} \left|\Sigma_g\right|^{0.5}} \exp\left(-0.5\left(x_n - \mu_g\right)' \Sigma_g^{-1} \left(x_n - \mu_g\right)\right). \tag{6}$$

The limited speaker data for teaching the speaker model is one of the main problems in speaker authentication. The universal background model is used for the initial quantification of the speaker model parameters. We adapt the universal model after its training to data from each speaker. We also use the expectation-maximization model for this purpose and for calculating the parameter averages, mixture weights, and the covariance matrix.

2.5. Baum-Welch Statistical Approximation

We use the universal background model parameters extracted by the expectation-maximization algorithm to estimate the Baum-Welch statistics of zero and the first order for each feature vector. The following Equation calculates the zero-order statistics of the i-th training data:

$$N_c\left(X_i\right)=\sum\gamma_{i,t}\left(c\right), \tag{7}$$

In which, $\gamma_{i,t}\left(c\right)$ is the a priori probability of the $X_{i,t}$ vector by the c-th component of the Gaussian mixture, which is expressed by:

$$\gamma_{i,t}\left(c\right)=\Pr\left(c\,|\,X_{i.t}\right)=\frac{w_c\Pr\left(X_{i.t}\,|c\right)}{\sum_{j=1}^{M}w_j\Pr\left(X_{i.t}\,|\,j\right)}. \tag{8}$$

Also, the first-order statistics of the c-th training data can be given as follows:

$$F_c\left(X_i\right)=\sum_i\gamma_{i,t}\left(c\right)\left(X_{i,t}-m_c\right), \tag{9}$$

In which, m_c is the average of the c-th Gaussian mixture component.

2.6. Training the Model and Super Vector Parameters

The following Equation will model a super vector that shows the features of a signal [16]:

$$M=m+Tw, \tag{10}$$

In which, m is a super vector independent of the speaker and channel made from the universal background model. Each super vector is made from putting together the component average vectors of the universal background model, which usually creates a rather long vector. Also, T shows the total variability matrix, which is a low-rank matrix. The latent w variable is the vector of total factors that is a randomized vector with a standard normal Gaussian distribution alongside an average of zero and unit covariance:

$$CN\left(0,I\right). \tag{11}$$

Notably, the a posteriori average vector of w is known as the identity vector. We also use the expectation-maximization algorithm to calculate the T matrix. The following Equation calculates the covariance matrix if the universal background model had c components and a feature vector dimension of D:

$$\Sigma=\begin{bmatrix}\Sigma_1 & \dots & 0\\ \mathrm{M} & \mathrm{O} & \mathrm{M}\\ 0 & \dots & \Sigma_c\end{bmatrix}, \tag{12}$$

In which, Σ_i is the covariance matrix of the c-th component from the universal background model. Also, x_i is the feature vector for training the i-th training data and $\Pr\left(x_i\,|\,M,\Sigma\right)$ shows the likelihood function of this function. Then, the expectation-maximization will be done in two stages:

1. It uses the T value to maximize the likelihood value of each training vector data. The following Equation calculates

$$w_i=\arg\max\Pr\left(X_i\,|\,m+Tw_i,\Sigma\right). \tag{13}$$

2. The following Equation calculates the updated T value:

$$\prod_i \Pr\left(X_i \mid m + Tw_i, \Sigma\right). \tag{14}$$

Then, we calculate the likelihood function of each signal as follows:

$$\sum_i \left(N_c \ln \frac{1}{(2\pi)^{M/2} \left|\Sigma_c\right|^{0.5}} - \frac{1}{2} \sum_t \left(X_{i\,t} - T_c w_i - m_c\right)' \Sigma_c^{-1} \left(X_{i\,t} - T_c w_i - m_c\right) \right) \tag{15}$$

In which, c and t relate to the feature model and vector components while T_c related to the c component. We calculate w_i as follows using the zero and first-order statistics of the a priori signal covariance matrix:

$$cow\ \left(w_i, w_i\right) = \left(I, \sum_i N_c\left(X_i\right) T_c{}' \Sigma_c^{-1} F_c\left(X_i\right) \right)^{-1} \tag{16}$$

Also, the average of the w_i variable equals:

$$E\left[w_i\right] = \operatorname{cov}\left(w_i, w_i\right) \sum_i T_c{}' \Sigma_c^{-1} F_c\left(X_i\right). \tag{17}$$

The second torque of the w_i variable equals to:

$$E\left[w_i, w_i'\right] = \operatorname{cov}\left(w_i, w_i\right) + E\left[w_i\right] E\left[w_i'\right]. \tag{18}$$

We use the following Equation to maximize Equation (13) and update the total variability matrix:

$$T_c = \left(\sum_i F_c\left(X_i\right) E\left[w_i'\right] \right) \times \left(\sum_i N_c\left(X_i\right) E\left[w_i, w_i'\right] \right)^{-1}. \tag{19}$$

2.7. Calculating the Identity-Vector

This uses the maximum a posteriori estimation of the w variable to calculate the identity vector. In reality, Equation (16) is the identity-vector Equation.

3. Identity Vector Normalization

The identity vector normalization methods are usually used to increase system performance. Uncorrelation transforms and whitening alongside intra-class covariance normalization algorithms are used to prepare data and reduce intra-class changes. Each of these methods is described in detail later on.

3.1. Uncorrelation Transforms and Whitening

This transform aims to uncorrelated and equalize the effects of features. It calculates the average and covariance of all identity vectors from the training set. Then, uses these values to uncorrelated and whiten all identity vectors. Finally, it maps the data it's the unit space [17].

3.2. Normalizing Intra-Class Correlation

This intra-class normalization aims to reduce the troublesome directions and intra-class changes just like the probabilistic linear discriminant analysis (PLDA) methods. This method is generally recommended to improve speaker authentication systems. This method must calculate the intra-class covariance matrix as follows at the beginning [18][19]:

$$s_w = \frac{1}{s}\sum_{s=1}^{s}\frac{1}{n_s}\sum_{i=1}^{n_s}\left(w_{s,i} - \bar{w}_s\right)\left(w_{s,i} - \bar{w}_s\right)', \tag{20}$$

In which, s is the overall number of classes, n_s is the number of samples in class s, $W_{s.i}$ is the i-th sample from class s, and $\bar{W}_s$ is the sample average from the s-th class. The WCCN transform matrix is calculated as follows using the Cholesky factoring on the intra-class covariance matrix:

$$s_w^{-1} = A_{wccn} \ A_{wccn}{}'. \tag{21}$$

Finally, the samples are calculated as follows in the new space:

$$\Phi_{wccn} = A'_{wccn} \ w \ , \tag{22}$$

Here, w is the identity vector and A'_{wccn} is the transfer matrix.

4. Simulation Results

This section presents the MATLAB simulation results to evaluate the proposed method's performance. Fig. 1 shows the proposed method for speaker authentication. Later on, we will explain the details regarding each section of this simulation.

4.1. System Configuration

We configure this system during the feature extraction stage using two independent methods as follows:

- Step One (MFCC): We use the hamming window with a width of 40ms and an overlap of 15s for the speech signal segmentation or framing. We reduced the TIMIT data set sampling frequency from 16 kHz to 8 kHz. Then, we implement the 26-part Mel Filter Bank on the TIMIT data set to extract the MFCC feature coefficients. This feature vector includes 19 main coefficients with their energy logarithm, first, and second derivatives to create a 60-dimensional feature vector.
- Step Two (DWT): We use the hamming window with a width of 40ms and an overlap of 15s for the speech signal segmentation or framing. We reduced the TIMIT data set sampling frequency from 16 kHz to 8 kHz. Then, the Wavelet coefficients are extracted from the TIMIT data set for each frame to create a 60-dimensional feature vector.
- Step Three: We connect the feature vectors from stages one and two as a series to gain a 120-dimensional feature vector. The first 60 arrays of this new vector are the extracted features from the first stage (MFCC) and the other 60 belong to the second stage (DWT).

A feature warping function turns the feature vector distribution of each file into a standard Gaussian distribution to reduce possible mismatches while normalizing the average and variance. We used an energy-based method to separate the speaking from silence and remove all silence frames or any extra information. Later on, we assumed the number of Gaussian mixture components for creating the universal background model as 256 to calculate the identity vectors. We came to this number by trial and error. We trained the total variability space matrix by the development data using the new universal background model and factor analysis to extract the identity vector of each audio file with 400 dimensions. Then, the probabilistic linear discriminant analysis reduces the dimensions of these identity vectors to 200. This probabilistic linear discriminant analysis reduces the calculation load, increases the inter-group changes, reduces intra-group changes, and increases the separability between speakers. This new vector distribution from the development data set is a good representation of vector distribution. Therefore, the registration and testing vectors are whitened by the statistical parameters of the development vectors. These vectors are completely uncorrelated because of the special covariance matrix vectors and values from the development data so we normalize or standardize them by turning their variance into one. Afterward, we create the model of each speaker by calculating the averages between their registration vectors.

4.2. Speech Signal Data Set

We use the TIMIT speech signal data in this paper. This data set includes speech samples from 630 speakers with eight different English accents (438 are men and 192 are women). This data set contains 10 short sentences for each speaker said in independent conditions. Each sentence is around 3s with phonetic diversity. This study only uses the speech samples of male speakers. We considered 368 speakers and each of their 10 sentences as the development data set to create the universal background model, the T matrix, LDA calculation, and PLDA teaching. We considered another 70 male speakers and 9 sentences from each for the registration phase alongside a 3s sentence for the test. There were a general number of 4900 authentication tests.

4.3. Non-Gaussian Noise Realization

We use the probability density function of the Gaussian mixture model to simulate real noise. The non-Gaussian model is much more accurate and complete compared to the Gaussian model because of numerous interferences or inaccurate calibrations; therefore, the importance of proper non-Gaussian noise modeling is one of the important usages of this paper. As was mentioned in Section 2.2., this paper considers the Gaussian mixture model as the noise probability density function. This Gaussian mixture model facilitates the exchange of many non-Gaussian noises with Gaussian mixtures. This paper uses three different non-Gaussian noise types to create a more accurate simulation.

1. We consider Equation (3) from Section 2.2 with a Gaussian component. This turns the non-Gaussian noise of Equation (3) into the following:

$$f_{\theta_1}\left(\theta_1\right) = N\left(0, 0.1\right), \tag{23}$$

Here, the $N\left(\mu, \sigma^2\right)$ is the Gaussian probability density function with an average of μ and variance of σ^2.

2. We consider the noise probability density function as a mixture of two Gaussian noises with an average of zero and different standard deviations. Therefore, Equation (3) will be rewritten as follows:

$$f_{\theta_2}\left(\theta_2\right) = 0.5\ N\left(0, 0.1\right) + 0.5\ N\left(0, 10\right), \tag{24}$$

3. We consider the noise probability density function as a mixture of three Gaussian noises with different averages and similar standard deviations. Therefore, Equation (3) will be rewritten as follows:

$$f_{\theta_3}\left(\theta_3\right)=0.2\ N\left(-3,0.1\right)+0.6\ N\left(0,0.1\right)+0.2N\left(3,0.1\right), \tag{25}$$

4.4. Hardware Impairment Realization

According to Section 3.2., η represents the distortion from non-ideal hardware in the speaker verification system, which is modeled as a Gaussian accumulative noise with an average of zero and variance of k^2 using Equation (2). In reality, this value determines the non-ideal hardware impairment in speaker verification systems. We consider three real simulations for hardware impairment realization.

1. Zero distortion () $k^2=0$: Equal to ideal hardware or lack of any hardware impairment in the speaker verification system. Therefore, Equation (2) will be as follows:
$$\eta_1 : N\left(0,k^2=0\right): 0 \tag{26}$$
2. Low distortion) $k^2=0.1$ (: Equal to low impairment in the speaker verification system. Therefore, Equation (2) will be as follows:
$$\eta_2 : N\left(0,0.1\right) \tag{27}$$
3. High distortion) $k^2=0.9$ (: Equal to high impairment in the speaker verification system. Therefore, Equation (2) will be as follows:

$$\eta_3 : N\left(0,0.9\right) \tag{28}$$

4.5. Results and Comparison with Other Methods

This section presents the proposed method results with different noises and hardware impairments alongside an equal error rate for the TIMIT data set in Tables 1 to 3. They were also compared to methods [6-8] if possible. We considered three different noises (1- Gaussian, 2- Non-Gaussian with Gaussian mixture model including two components and equal averages, 3- Non-Gaussian model with a Gaussian mixture model including three components and equal variances) and three different hardware impairments (1- Ideal hardware, 2- Low impairment rate, 3- High impairment rate), which makes a total of 9 different combinations compared in three tables. We claim our paper to be one of the first studies that mix the hardware impairment and non-Gaussian noise fields; therefore, we cannot conduct a fair comparison with methods [6-8]. Later on, we compare the proposed method with different time-frequency methods. We also consider the following four scenarios to have a comprehensive simulation of speaker authentication systems:

- Gaussian noise without hardware impairment
- Non-Gaussian noise without hardware impairment
- Gaussian noise with low and high impairment
- Non-Gaussian noise with low and high impairment

4.5.1. Gaussian Noise + Zero Hardware Impairment

We can compare this special scenario of Gaussian noise (or $f_{\theta_1}\left(\theta_1\right)=N\left(0,0.1\right)$) without any hardware impairment (an equivalent $f_{\eta_1}\left(\eta_1\right): 0$, which signifies an ideal system) with many other studies. This proposed system with its DWT-MFCC feature extraction algorithm has the best performance compared to other methods such as time-frequency methods. The first row of Table 1 shows a summary of these results.

4.5.2. Non-Gaussian Noise + Zero Hardware Impairment

- The proposed method has a better performance compared to the study [6] and other time-frequency methods in the special scenario of non-Gaussian noise including two Gaussian components with different standard deviations or its equivalent $f_{\theta_2}\left(\theta_2\right)=0.5\;N\left(0,0.1\right)+0.5\;N\left(0,10\right)$, and no hardware impairment.

  The first row of Table 2 shows a summary of these results.
- The proposed method has a better performance compared to the study [6] and other time-frequency methods in the special scenario of non-Gaussian noise including three Gaussian components with different averages or its equivalent $f_{\theta_3}\left(\theta_3\right)=0.2\;N\left(-3,0.1\right)+0.6\;N\left(0,0.1\right)+0.2N\left(3,0.1\right)$, and no hardware impairment. The first row of Table 3 shows a summary of these results.

4.5.3. Gaussian Noise + Low and High Impairment Rates

We cannot compare this scenario with studies [6-8] so we only presented a comparison between the proposed method and other time-frequency methods. The DWT-MFCC had the best results among all time-frequency methods. Also, increasing the hardware impairment rate increases the equal error rate. The second and third rows of Table 1 show a summary of these results.

4.5.4. Non-Gaussian Noise + Low and High Impairment Rates

The comparison of this scenario with other time-frequency methods such as Gabor-MFCC, STFT-MFCC, and Winger-Ville-MFCC show the better performance of the proposed method. Also, increasing the hardware impairment rate considerably increases the equal error rate. The second and third rows of Tables 2 and 3 show a summary of these results.

Figure 2. shows the bivariate histogram of the equal error rate (EER) for the proposed method, i.e., DWT-MFCC. The horizontal axis indicates different noise models and the vertical axis indicates the number of hardware impairments. In Figure 3, for different noises and hardware impairments, the EER varies from 0.1 to 0.4, so that the lowest EER value (i.e., 0.1) occurs in the case of Gaussian noise and zero hardware impairments, which is marked in blue. And the highest amount of error (i.e., 0.4) occurs in the case of non-Gaussian noise and high hardware impairment, which is marked in brown.

5. Conclusion

This paper analyzed the effects of hardware impairment and non-Gaussian noise on the performance of speaker authentication systems. The mixed DWT-MFCC extracted the features of the TIMIT data set to implement the system in the identity vector space. We used PLDA to improve the system performance. The MATLAB test results showed that increasing the hardware impairment rate also increases the widely used equal error rate. Also, the simultaneous existence of non-Gaussian noise and hardware impairment considerably reduce the system performance.

References

[1] J. H. L. Hansen and T. Hasan, "Speaker Recognition by Machines and Humans: A tutorial review," Signal Processing Magazine, IEEE, vol. 32, pp. 74-99,2015.

[2] E. Lleida and L. J. Rodriguez-Fuentes, "Speaker and language recognition and characterization: Introduction to the CSL special issue," ed: Elsevier, 2018.

[3] X. Guo, Y. He, S. Atapattu, S. Dey, and J. S. Evans," Power Allocation for Distributed Detection Systems in Wireless Sensor Networks With Limited Fusion Center Feedback," inIEEE Transactions on Communications, vol. 66, no. 10, pp. 4753-4766, Oct.2018.

[4] G. Ding, X. Gao, Z. Xue, Y. Wu and Q. Shi, "Massive MIMO for Distributed Detection With Transceiver Impairments," in IEEE Transactions on Vehicular Technology, vol. 67, no.1, pp. 604-617, Jan. 2018.

[5] R. Annavajjala, C. C. Yu and J. M. Zagami, "Communication over non-Gaussian channels - Part I: Mutual information and optimum signal detection," MILCOM 2015 - 2015 IEEE Military Communications Conference, Tampa, FL, 2015, pp. 1126-1131.

[6] S Al-Kaltakchi, Musab T., et al. "Evaluation of a speaker identification system with and without fusion using three databases in the presence of noise and handset effects." EURASIP Journal on Advances in Signal Processing 2017.1 (2017): 1-17.

[7] Dehak, Najim, Patrick J. Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet. "Front-end factor analysis for speaker verification." IEEE Transactions on Audio, Speech, and Language Processing 19, no. 4 (2010): 788-798.

[8] M. T. S. Al-Kaltakchi, R. R. O. Al-Nima, M. Alfathe and M. A. M. Abdullah, "Speaker Verification Using Cosine Distance Scoring with i-vector Approach," 2020 International Conference on Computer Science and Software Engineering (CSASE), 2020, pp. 157-161.

[9] X. Zhang, X. Zou, M. Sun, T. F. Zheng, C. Jia and Y. Wang, "Noise Robust Speaker Recognition Based on Adaptive Frame Weighting in GMM for i-Vector Extraction," in IEEE Access, vol. 7, pp. 27874-27882, 2019.

[10] Al-Kaltakchi, M.T.S., Al-Nima, R.R.O., Abdullah, M.A.M. et al. "Thorough evaluation of TIMIT database speaker identification performance under noise with and without the G.712 type handset". Int J Speech Technol 22, 851–863 (2019).

[11] Kinnunen, T. and H. Li, "An overview of text-independent speaker recognition: From features to supervectors." Speech communication, 52(1): p. 12-40. 2010.

[12] Yang, H.-T. and C.-C. Liao, "A de-noising scheme for enhancing wavelet-based power quality monitoring system" Transactions on Power Delivery, 16(3): p. 353-360. 2001.

[13] Sahidullah, Md.; Saha, Goutam (May 2012). "Design, analysis and experimental evaluation of block based transformation in MFCC computation for speaker recognition". Speech Communication. 54 (4): 543–565.

[14] Reynolds, D.A., T.F. Quatieri, and R.B. Dunn, "Speaker verification using adapted Gaussian mixture models." Digital signal processing, 10(1-3): p. 19-41. 2000.

[15] Kenny, P., et al., "A study of interspeaker variability in speaker verification." IEEE Transactions on Audio, Speech, and Language Processing, 16(5): p. 980. 2008.

[16] Ghahabi, O. and J. Hernando, "I-vector modeling with deep belief networks for multi-session speaker recognition. " network, 20:p.13.2014.

[17] Greenberg, C.S., et al. "The NIST 2014 speaker recognition i-vector machine learning challenge." in Odyssey: The Speaker and LanguageRecognitionWorkshop.2014.

[18] Hatch, A.O., S. Kajarekar, and A. Stolcke. "Within-class covariance normalization for SVM-based speaker recognition." in Ninth international conference on spoken language processing. 2006.

[19] Solomonoff, A., W.M. Campbell, and I. Boardman. "Advances in channel compensation for SVM speaker recognition. in Acoustics, Speech, and Signal Processing," 2005. Proceedings.(ICASSP'05). IEEE International Conference on. 2005.IEEE.

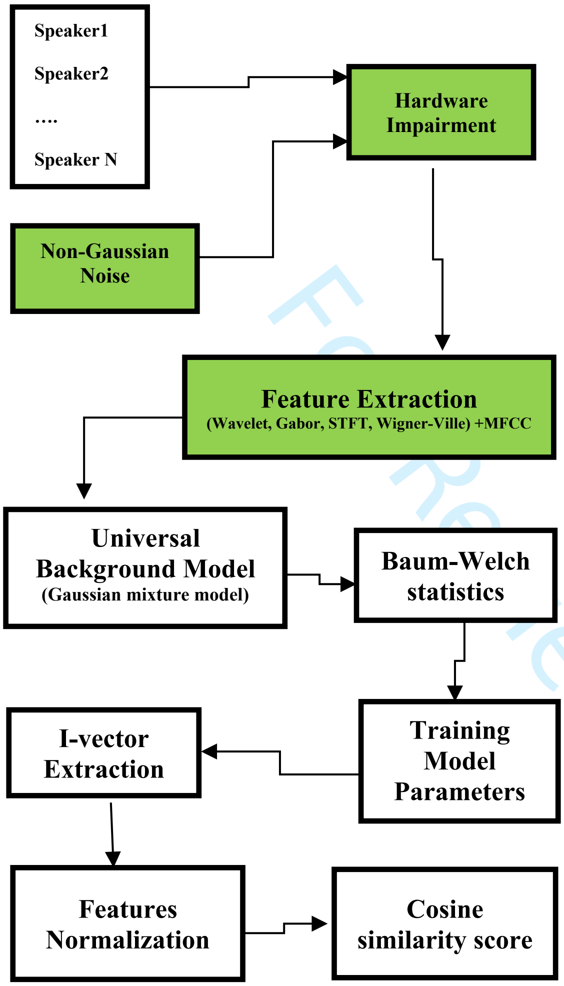


Fig. 1. Proposed (applicable) block diagram for speaker authentication.

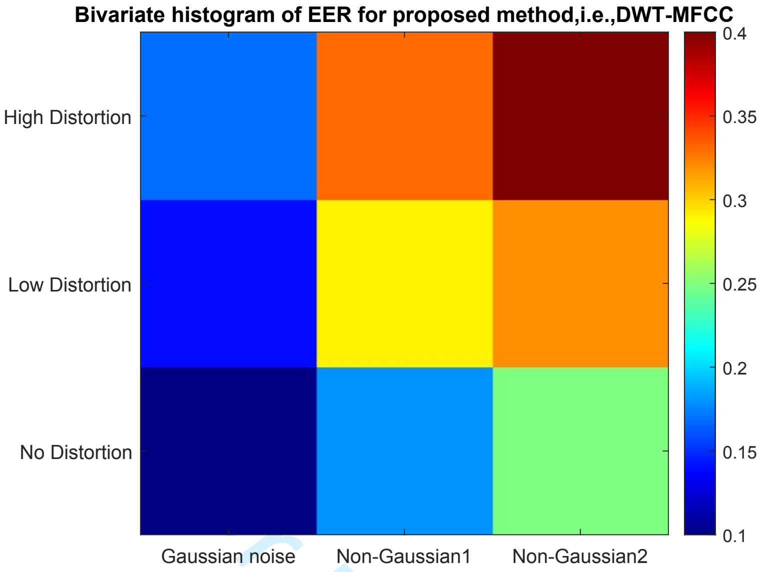


Fig. 2. Bivariate histogram of the EER for DWT-MFCC method in different noise models and hardware impairments.

Table 1. Equal error rate point for Gaussian noise and different hardware impairment rates (ideal hardware, low impairment, high impairment) of the proposed and other methods.

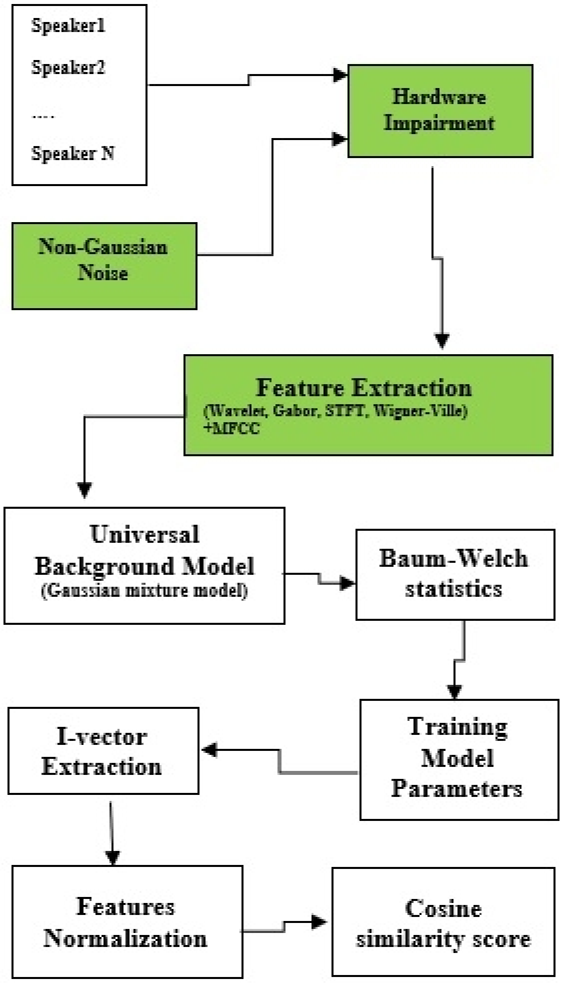
		Equal Error Rate						
Hardware impairment	Noise	DWT-MFCC	STFT-MFCC	Gabor-MFCC	Wigner-Ville-MFCC	Method[6]	Method[7]	Method[8]
No distortion(Eq. 25)	Gaussian(Eq. 22)	**0.10**	0.12	0.15	0.12	0.13	0.16	0.18
Low distortion(Eq. 26)	Gaussian(Eq. 22)	**0.14**	0.18	0.20	0.18	---	---	---
High distortion(Eq. 27)	Gaussian(Eq. 22)	**0.17**	0.20	0.23	0.21	---	---	---

Table 2. Equal error rate point for non-Gaussian noise with a mixture of two Gaussian components and different hardware impairment rates (ideal hardware, low impairment, high impairment) of the proposed and other methods.

		Equal Error Rate						
Hardware impairment	Noise	DWT-MFCC	STFT-MFCC	Gabor-MFCC	Wigner-Ville-MFCC	Method[6]	Method[7]	Method[8]
No distortion (Eq. 25)	Non-Gaussian(Eq.23)	**0.18**	0.21	0.24	0.19	0.23	---	---
Low distortion (Eq. 26)	Non-Gaussian(Eq.23)	**0.29**	0.32	0.34	0.32	---	---	---
High distortion (Eq. 27)	Non-Gaussian(Eq.23)	**0.33**	0.42	0.46	0.41	---	---	---

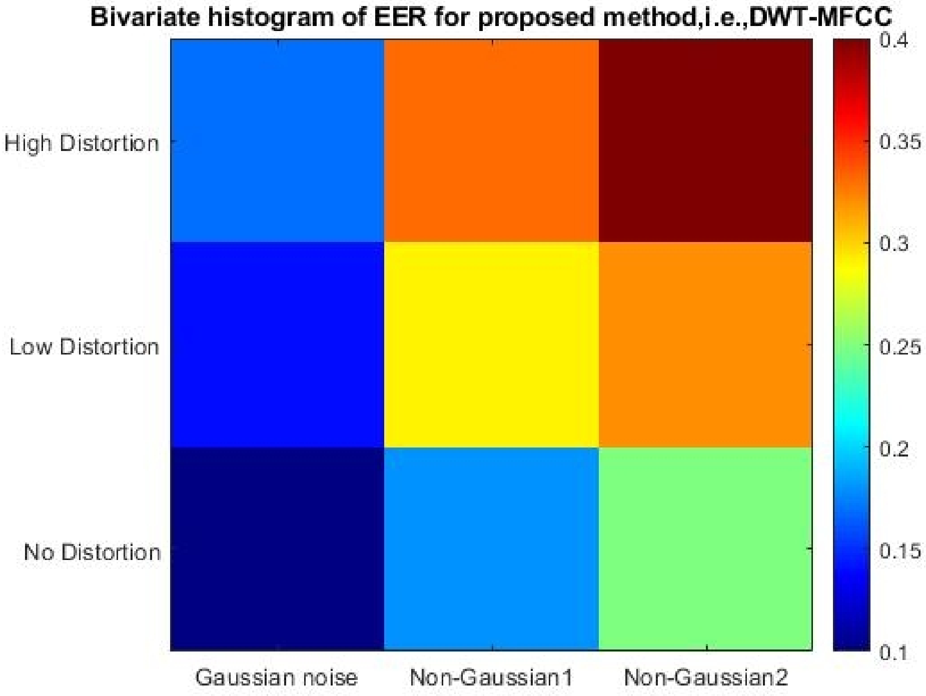
Table 3. Equal error rate point for non-Gaussian noise with a mixture of three Gaussian components and different hardware impairment rates (ideal hardware, low impairment, high impairment) of the proposed and other methods.

		Equal Error Rate						
Hardware impairment	Noise	DWT-MFCC	STFT-MFCC	Gabor-MFCC	Wigner-Ville-MFCC	Method[6]	Method[7]	Method[8]
No distortion (Eq. 25)	Non-Gaussian(Eq.24)	**0.25**	0.29	0.36	0.28	0.35	---	---
Low distortion (Eq. 26)	Non-Gaussian(Eq.24)	**0.32**	0.38	0.41	0.38	---	---	---
High distortion (Eq. 27)	Non-Gaussian(Eq.24)	**0.40**	0.44	0.48	0.45	---	---	---



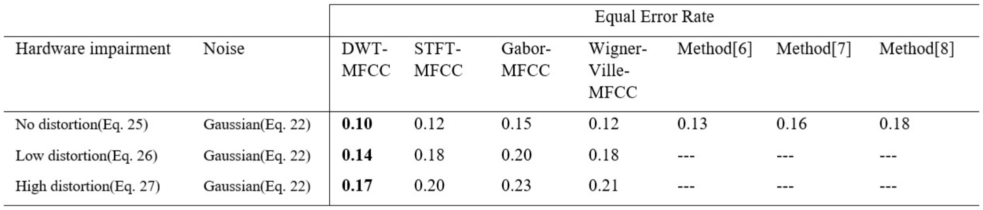
Proposed (applicable) block diagram for speaker authentication.

78x132mm (96 x 96 DPI)



Bivariate histogram of the EER for DWT-MFCC method in different noise models and hardware impairments.

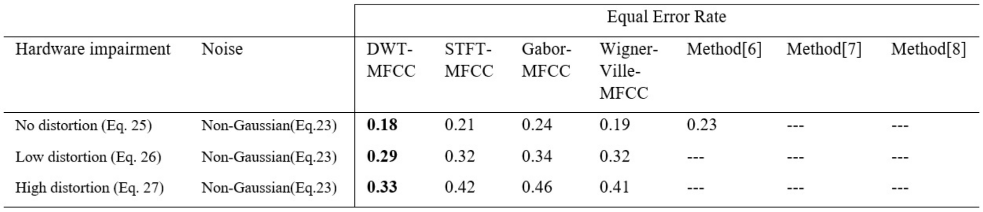
197x148mm (72 x 72 DPI)



		Equal Error Rate						
Hardware impairment	Noise	DWT-MFCC	STFT-MFCC	Gabor-MFCC	Wigner-Ville-MFCC	Method[6]	Method[7]	Method[8]
No distortion(Eq. 25)	Gaussian(Eq. 22)	**0.10**	0.12	0.15	0.12	0.13	0.16	0.18
Low distortion(Eq. 26)	Gaussian(Eq. 22)	**0.14**	0.18	0.20	0.18	---	---	---
High distortion(Eq. 27)	Gaussian(Eq. 22)	**0.17**	0.20	0.23	0.21	---	---	---

Equal error rate point for Gaussian noise and different hardware impairment rates (ideal hardware, low impairment, high impairment) of the proposed and other methods.

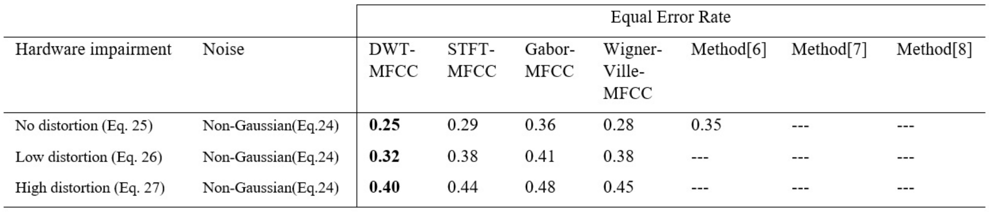
336x80mm (96 x 96 DPI)



		Equal Error Rate						
Hardware impairment	Noise	DWT-MFCC	STFT-MFCC	Gabor-MFCC	Wigner-Ville-MFCC	Method[6]	Method[7]	Method[8]
No distortion (Eq. 25)	Non-Gaussian(Eq.23)	**0.18**	0.21	0.24	0.19	0.23	---	---
Low distortion (Eq. 26)	Non-Gaussian(Eq.23)	**0.29**	0.32	0.34	0.32	---	---	---
High distortion (Eq. 27)	Non-Gaussian(Eq.23)	**0.33**	0.42	0.46	0.41	---	---	---

Equal error rate point for non-Gaussian noise with a mixture of two Gaussian components and different hardware impairment rates (ideal hardware, low impairment, high impairment) of the proposed and other methods.

336x84mm (96 x 96 DPI)



		Equal Error Rate						
Hardware impairment	Noise	DWT-MFCC	STFT-MFCC	Gabor-MFCC	Wigner-Ville-MFCC	Method[6]	Method[7]	Method[8]
No distortion (Eq. 25)	Non-Gaussian(Eq.24)	**0.25**	0.29	0.36	0.28	0.35	---	---
Low distortion (Eq. 26)	Non-Gaussian(Eq.24)	**0.32**	0.38	0.41	0.38	---	---	---
High distortion (Eq. 27)	Non-Gaussian(Eq.24)	**0.40**	0.44	0.48	0.45	---	---	---

Equal error rate point for non-Gaussian noise with a mixture of three Gaussian components and different hardware impairment rates (ideal hardware, low impairment, high impairment) of the proposed and other methods.

334x87mm (96 x 96 DPI)