IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 22, 2017, accepted June 16, 2017, date of publication June 21, 2017, date of current version July 24, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2717842

Blind Detection of Copy-Move Forgery in Digital Audio Forensics

MUHAMMAD IMRAN[1], ZULFIQAR ALI[1], SHEIKH TAHIR BAKHSH[2], AND SHEERAZ AKRAM[3]

[1]College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia
[2]Computer Science Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[3]Department of Software Engineering, Foundation University, Islamabad 44000, Pakistan

Corresponding authors: Muhammad Imran (cimran@ksu.edu.sa) and Zulfiqar Ali (zuali@ksu.edu.sa)

This work was supported by the Deanship of Scientific Research of King Saud University, Riyadh Saudi Arabia through the Research Group, under Project RG-1435-051.

ABSTRACT Although copy-move forgery is one of the most common fabrication techniques, blind detection of such tampering in digital audio is mostly unexplored. Unlike active techniques, blind forgery detection is challenging, because it does not embed a watermark or signature in an audio that is unknown in most of the real-life scenarios. Therefore, forgery localization becomes more challenging, especially when using blind methods. In this paper, we propose a novel method for blind detection and localization of copy-move forgery. One of the most crucial steps in the proposed method is a voice activity detection (VAD) module for investigating audio recordings to detect and localize the forgery. The VAD module is equally vital for the development of the copy-move forgery database, wherein audio samples are generated by using the recordings of various types of microphones. We employ a chaotic theory to copy and move the text in generated forged recordings to ensure forgery localization at any place in a recording. The VAD module is responsible for the extraction of words in a forged audio, and these words are analyzed by applying a 1-D local binary pattern operator. This operator provides the patterns of extracted words in the form of histograms. The forged parts (copy and move text) have similar histograms. An accuracy of 96.59% is achieved, and the proposed method is deemed robust against noise.

INDEX TERMS Digital multimedia forensics, audio forgery, authentication, blind detection, copy-move forgery.

I. INTRODUCTION

Digital audio authentication/forensics is attracting growing attention from information security researchers [1] because of various reasons, such as ingenious and convenient forgery techniques (e.g., insertion, copy-move, splicing), smart editing tools (e.g., Audio Audition), unauthentic sources (e.g., web and social networks), and a wide range of applications in different domains. For example, a pre-recorded audio conversation can be forged for monetary gains, access control, and as an alibi in the court of law [2]. One or more parts of an audio message can be easily changed, mingled, or may not belong to the original speaker. In such cases, verification of audio integrity (i.e., authentication), as well as forgery detection and localization (i.e., forensics), are of paramount concern. Generally, it involves audio acquisition, analysis, and evaluation to determine its originality and possible manipulations. In the past, audio authentication was simple and convenient to investigate through spectrograms because they usually expose the irregularities and abrupt changes due to tampering (e.g., environment, speaker, contents), which can be observed from Fig. 1. However, recent technological advancements and the availability of advanced tools have made the detection of any type of forgery nearly impossible through hearing or visual analysis. Several sophisticated tools for manipulation and tampering of recording are widely available [3], [4] and can be easily employed for audio fabrication without leaving a telltale. Therefore, understanding the nature of audio forgery is crucial before detection or localization.

Audio recordings can be fabricated by applying splicing or copy-move because both types of forgeries are considered as the most dangerous manipulation methods for digital media tampering [5]. In the former, two or more recordings are used to generate forged audio; in the latter, some part of an audio is copied and moved in the same recording at different locations. Most of the existing forgery detection techniques can be categorized into active or passive (blind). Active techniques rely on embedding and extracting a

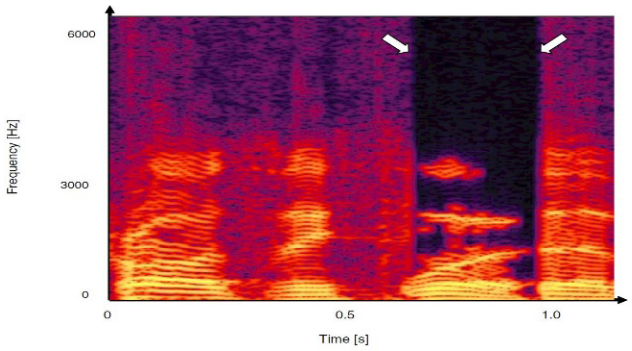

FIGURE 1. Spectrogram exposing abrupt changes due to insertion of an audio segment of different environments [6].

watermark in the audio to determine its originality. The audio is original if the embedded and the extracted watermarks are the same, and forged otherwise [7]. However, a watermark in most real-life scenarios is unknown; hence, such techniques become impractical. By contrast, passive techniques [8], [9] are considered more practical because they do not rely on any signature or watermark. However, audio forgery detection through blind techniques (i.e., passive) is a difficult task. Moreover, localization of a tampered region in a forged audio is more challenging compared to a binary decision (i.e., original or forged). The main objective of this study is to present a highly accurate passive copy-move forgery detection and localization system for audio.

Unlike copy-move, most of the existing literature [8]–[11] is geared towards splicing audio forgery detection. This is mainly because it is relatively easier to identify recordings of different environments, microphones, and speakers that were used to generate a forged audio. For example, the authors in [12] implemented MFCC and MPEG-7 based features to recognize various environments. Similarly, a very first approach to classifying microphones was presented in [13] to determine audio authentication. Various approaches were proposed for the classification of microphones and recording devices to authenticate an audio in [14]–[17]. Splicing forgery may also involve combining recordings of different speakers. Various approaches [18]–[22] were presented for speaker identification or recognition to cope with such forgery. We limit our discussion to copy-move forgery afterward because this paper deals with the same problem.

To the best of our knowledge and literature review, copy-move forgery detection either through active or passive techniques is still in its infancy, and a very limited work has been conducted with regard to audio. For example, Yan *et al.* [23] used pitch similarity to detect copy-move forgery in an audio. A robust pitch tracking algorithm YAAPT [24] was implemented to extract the pitch sequence of all the words of an audio recording. The experiments concluded that the pitch sequence of the original word and its copy were exactly the same. Different similarity measures, such as Pearson correlation coefficient and average difference, were computed to compare the pitch sequence. To the best of our effort, we were unable to find any other work that tackles the same problem.

In this paper, we present a novel method for blind detection and localization of a copy-move forgery for digital audio authentication and forensics. The proposed method employs a sophisticated voice activity detection (VAD) module to determine the boundary points of each uttered word in an audio and calculates corresponding histograms to observe the similarity between the original and the audio in question. Histograms are computed with the one-dimensional local binary pattern operator. Histograms for the two words are exactly the same; therefore, the proposed method can easily detect and locate the original word and its copy located at a different place in the audio. A copy-moved forged audio database is also created in this study to evaluate the performance of the proposed method. The database is generated in such a cultured way that no human can judge a recording by hearing and visualization. The location of copy-move forgery can be at any place in the recording. Therefore, chaotically generated random numbers are used to create the locations for copying and moving the text in recordings. The forged samples are generated by using the audios recorded in an office, cafeteria, and a soundproof room to ensure that the proposed method can work for different recording environments and devices. The recording equipment, including sound cards and microphone, varies for these environments. Microphones, also known as sensors or acoustic to electric transducer, differ from each other in polar patterns, which signifies the sensitivity of a microphone to sound arriving from different directions to its central axis. All audio samples of the generated forged database are used to evaluate the performance of the proposed method. In 96.59% of the audio samples, copy-move forgery is perfectly detected. A detection error of 3.41% appeared due to the inaccurate estimation of the boundary points.

The rest of the paper is organized as follows. Section 2 describes the detailed process for generation of copy-move forged database. The proposed system for the localization of copy-move forgery in an audio recording is also explained in the same section. Section 3 provides the experimental setup and the results of the proposed method by using the generated database. Section 4 shows the robustness of the proposed method against the noise. The analysis and comparison with the existing studies are also provided in Section 4. Finally, Section 5 provides a few conclusions.

II. PROPOSED METHOD FOR BLIND DETECTION OF COPY-MOVE FORGERY

In this section, the process to generate the copy-move forged audio database and components of the proposed methods are described.

A. GENERATION OF COPY-MOVE FORGED AUDIO DATABASE

A copy-move forged speech database is developed to evaluate the performance of the proposed method. Recording of various environments and equipment are considered to generate

the forged audio. The VAD module is implemented in such a way that forged recording should not be judged by any human being. The location of forgery could be at any place in a recording; therefore, the locations for copying and moving the text are generated through chaotic theory.

1) AUDIO RECORDINGS FOR COPY-MOVE FORGED DATABASE

Audio recordings of King Saud University Arabic Speech Database (KSUD) [25] are used to generate copy-move forged database. The database is available through the Linguistic Data Consortium, which is hosted by the University of Pennsylvania, Philadelphia, USA. KSUD has diversity in many aspects [26], [27]. Speakers of 29 nationalities from Arab, African, and Indian subcontinents were recorded in KSUD in three different sessions. In the first session, the recording was conducted in three different environments: soundproof room, office, and cafeteria. However, in the last two Sessions, the cafeteria was not included to reduce the recording time of a speaker. The recordings of the soundproof room represent a quiet environment, whereas the recordings in the office contain some background noise. Moreover, recordings in the cafeteria have people and background noise. Every speaker recorded various texts, including digits, sentences, paragraphs, and a question-and-answer session. The text by every speaker in each environment and session was recorded using diverse equipment, which differs from each other in terms of microphone quality and sound cards.

To the best of our knowledge, KSUD is one of the most comprehensive databases due to a variety of recording environments, equipment, and speakers. The block diagram to generate the copy-move database is depicted in Fig. 2. The recordings of digits in all three environments are processed to develop a copy-move forged database. For all environments, there are many options of recording equipment. For the sake of diversity, we consider the recordings of different sound cards. Moreover, microphones used in the office and cafeteria are similar; however, they are different from the soundproof room. The selected environments with recording equipment to produce the copy-move forged database are:

- In the soundproof room, a Shure (Beta 58A) microphone is coupled with a Yamaha sound mixture (MW-12CX).
- In the office, a Sony (F-V220) microphone is linked with an external sound card (Sound Blaster X-Fi Surround 5.1 Pro).
- In the cafeteria, a Sony (F-V220) microphone is connected with a built-in sound card of the desktop (OptiPlex 760).

The sensors Shure Beta 58A and Sony F-220 have different polar patterns. The pattern of sensor Shure Beta is super cardioid, and this pattern is used for sound reinforcement in conditions where noise is a major problem. By contrast, the pattern for Sony F-V220 is omnidirectional and senses the sound from all directions with equal gain, unlike the super cardioid pattern that picks up sounds with high gain from the front and sides but poorly from the rear.

2) VOICE ACTIVITY DETECTION MODULE

A voice detection module [28], [29] is one of the most crucial components in the development of a copy-move forged database because this module is responsible for the accurate detection of boundary points of a text in an audio. In a copy-move forged audio, a text from location i is copied and moved to some location j. If boundary points are not accurately determined, then a few parts of the copied text can either be missed or may contain silence. When this inaccurately copied text is moved to some other location, the audio can be easily identified as forged. Therefore, we deliberated and implemented the VAD module by considering various measures, such as total amplitude, zero-crossing (ZC), and duration of a text.

An audio is divided into successive non-overlapping frames $[A_1, A_2, A_3, \ldots, A_e, \ldots, A_t]$ to calculate these measures. Each of these frames has a duration of 20 milliseconds and k number of samples. The duration of frames is kept small; hence, it can be easily ignored in the case of silence. The total amplitude T_e for a frame e is calculated by using

$$T_e = \sum_{s=1}^{k} |a_s|, \tag{1}$$

where $[a_1, a_2, a_3, \ldots, a_k]$ are amplitudes in frame e. Total amplitude of each frame is computed and used to differentiate between voiced and unvoiced segments of an audio. A threshold *thresh* provided by Eq. (2) is used to determine the voiced frames. If total amplitude T of a frame A_e is greater than the *thresh*, then it is a voiced segment; otherwise, it is an unvoiced segment.

$$thresh = 3\% \text{ of } [\max(T) - \min(T)] + \min(T), \tag{2}$$

where $T = [T_1, T_2, T_3, \ldots, T_t]$, and $\max(T)$ and $\min(T)$ represent the maximum and minimum total amplitude in an audio, respectively.

The second measure is ZC, which becomes very high in the silence parts of an audio due to background noise. In the implemented VAD module, the ZC for silence segments of an audio is made zero by subtracting a certain amplitude from an audio. This amplitude is measured by using Eq. (2). By doing so, the amplitude of the silence parts in an audio will become negative; hence, zero ZC will exist. The last measure is the duration of a text. The VAD module may split a text into two segments, *segX* and *SegY*, because of a short pause inside the text. The following condition is used to avoid this situation,

$$\begin{aligned} &if\,[\text{duration}\,(SegX, SegY)\ AND\ silence(SegX, SegY)] \\ &\qquad < 0.3sec.\ then\ \text{merge}\,(SegX, SegY). \end{aligned} \tag{3}$$

The duration of a digit and silence between consecutive digits is approximately 0.5 and 0.4 seconds, respectively. Therefore, if the duration of split segments and the silence between them is less than 0.3 seconds, then they are merged because they are part of the same digits as mentioned in Eq. (3).

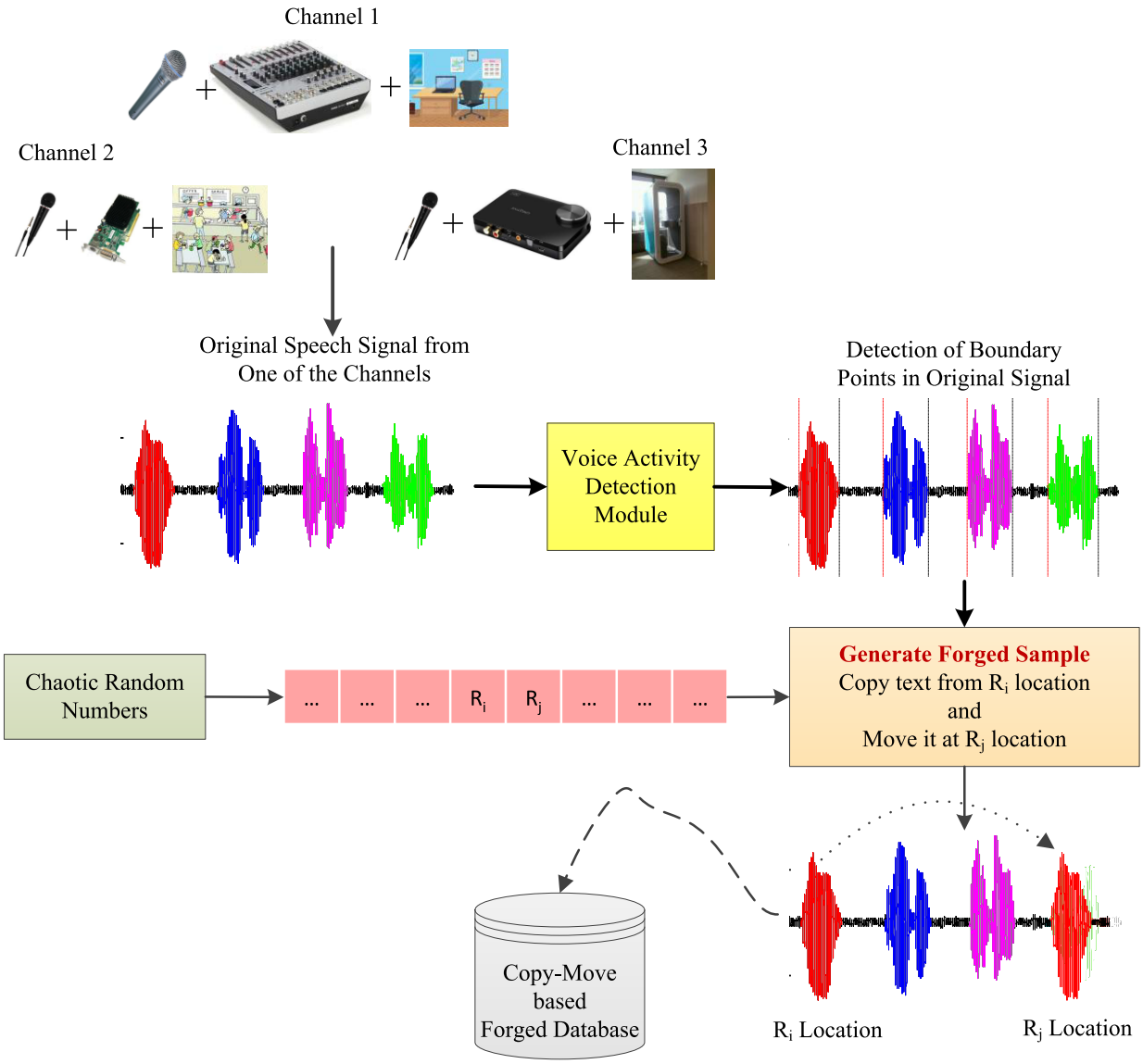

FIGURE 2. The process for generation of copy-move forged database.

3) CHAOTIC THEORY AND COPY-MOVE OF TEXT

After detection of boundary points, the next important step is to determine the locations to copy and move the text to generate forged audio. The locations of copy and move are determined by using chaotic theory. Unlike randomness, chaotic behavior is deterministic, and the numbers generated through the chaotic system can be regenerated by using the same initial conditions. We have used Gingerbreadman chaos theory [30] to determine the locations. The theory is defined by two-dimensional piecewise linear transformation and is provided in Eqs. (4) and (5) as follows:

$$x_{r+1} = 1 - y_r + |x_r|, \tag{4}$$

$$y_{r+1} = x_r, \tag{5}$$

where x_o and y_o are the initial conditions. In the generated sequence y, every two consecutive numbers can be used to determine the locations to copy and move the text. By using these locations, any number of copy-move forged samples can be generated from an original audio. For example, if a recording contains 10 parts of a text and we want to generate three forged samples from this text, then a sub-sequence y_c of y containing six numbers can be scaled between one and ten. Every two consecutive numbers will describe the locations to copy and move the text for a forged sample. The sub-sequence y_c can be scaled in the range from u to v by using

$$Y = u + \left\lfloor (v-u) \times \left(\frac{y_c - \min(y_c)}{\max(y_c) - \min(y_c)} \right) \right\rfloor. \tag{6}$$

When the location to copy and move the text is decided, say i and j, the text is copied by using the boundary points of

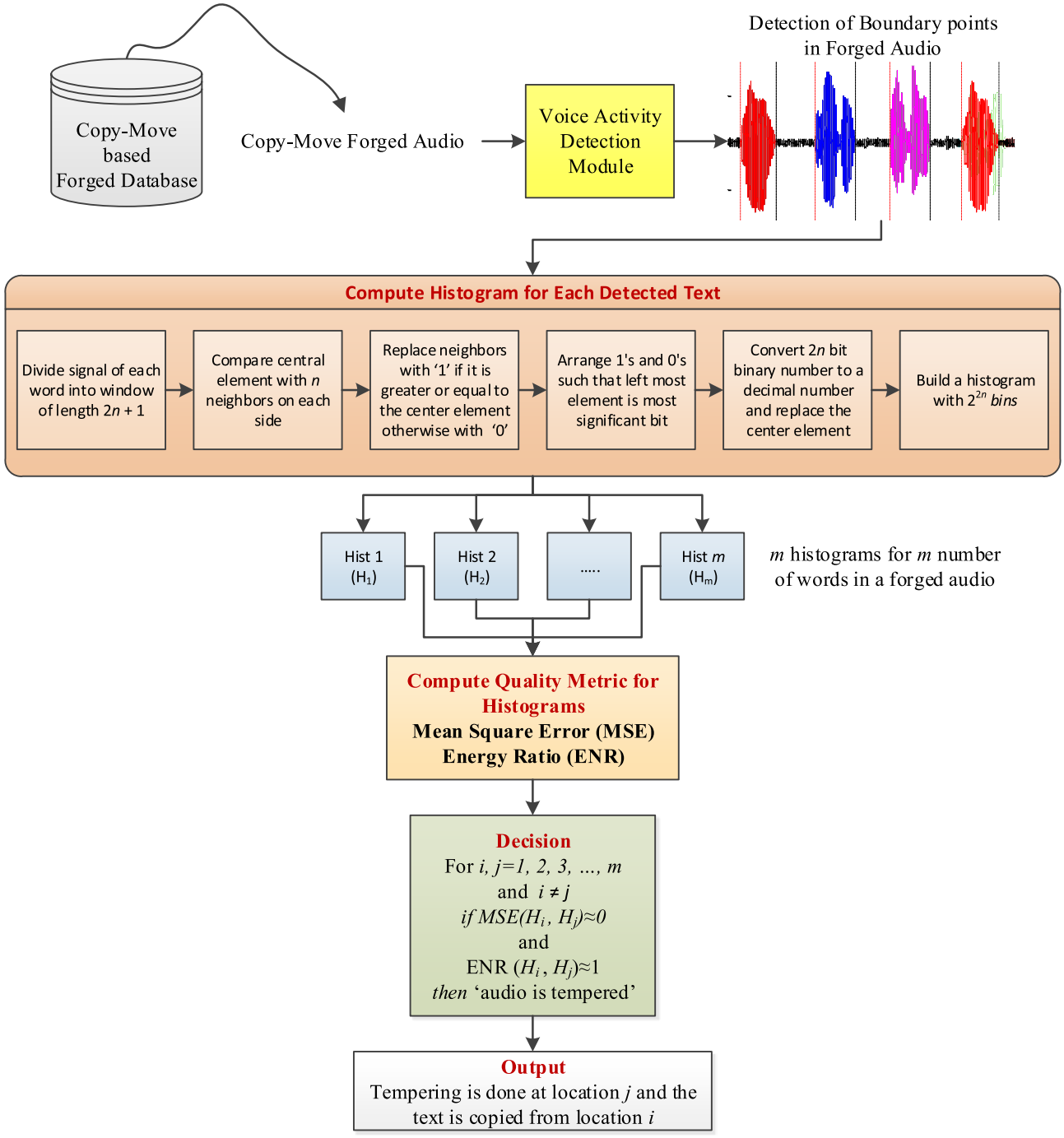

FIGURE 3. An overview of the proposed copy-move forgery detection system.

the text at location i. Now, the text at location j is removed by using its boundary points, and the copied text is placed at location j. Thus, a forged copy-move recording is generated.

B. THE PROPOSED METHOD TO DETECT COPY-MOVE FORGERY

A method to detect copy-move forgery is proposed in this study. The overview of the proposed system is presented in Fig. 3. The first step to detect the forgery is the detection of boundary points of the forged audio by implementing the VAD module. During generation of forged audio, the text at location j is removed and copied text from location i is placed at location j. In such case, if the text at location i and j have different durations, then the total duration of the forged audio becomes different than the original audio. Therefore, implementing the VAD module again is necessary to detect

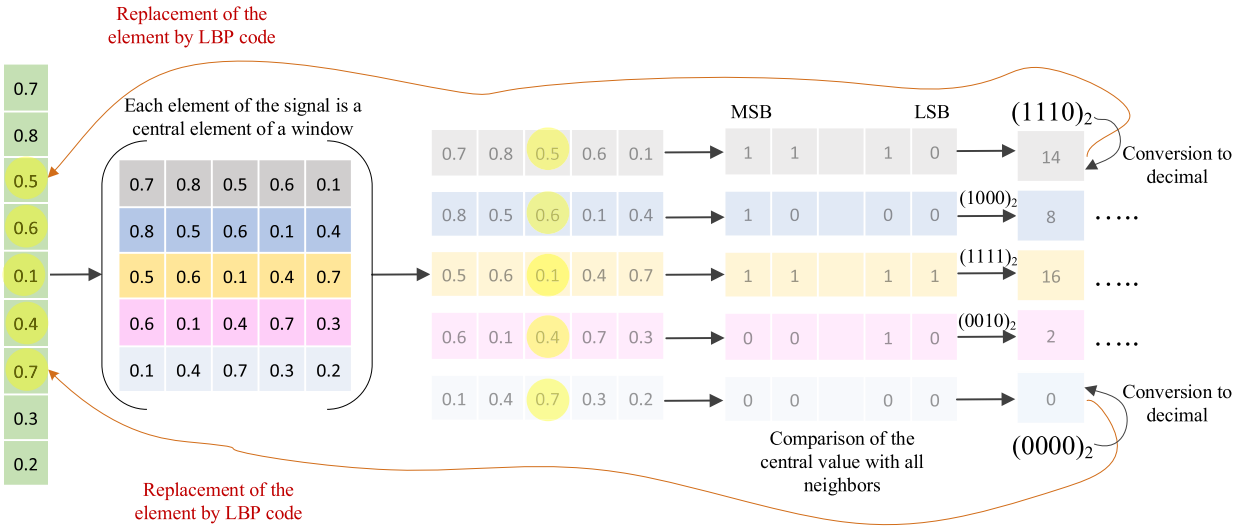

FIGURE 4. Computation of LBP codes.

the boundary points in a forged audio. Two major steps of the proposed method are described in the following subsections. These components include the calculation of histograms for all words of a forged audio and their comparisons with each other to determine the forgery location.

1) COMPUTATION OF HISTOGRAMS FOR ALL TEXTS

Consider an audio U_{CM} from a developed copy-move forged database. A histogram is computed for each word in the audio to detect the location and contents of forgery, and the local binary pattern (LBP) is implemented for this purpose. The first step in LBP is a division of forged audio U_{cm} into small windows such that each sample of U_{CM} is a center of a window. The length of each window is $2n + 1$, where n represents the number of neighbors on each side of a center element. The center element of each window will be replaced by an LBP code. In the case of N number of samples in U_{CM}, the total numbers of windows are $N - 2n$ when no zero padding is done on either side of U_{CM}. The windows of U_{CM} are provided by Eq. (7)

$$U_{CM} = \begin{bmatrix} u_1 & u_2 & \cdots & u_{2n+1} \\ u_2 & u_3 & \cdots & u_{2n+2} \\ \vdots & \vdots & \ddots & \vdots \\ u_{N-4n} & u_{N-4n+1} & \cdots & u_{N-2n} \end{bmatrix}. \tag{7}$$

The center element u_{n+1} of the window will be compared with each of its neighbors on both sides to compute the LBP code for a given window $u_1, u_2, \ldots, u_{n+1}, \ldots, u_{2n+1}$ of length $2n + 1$. If neighbors are greater or equal to the center element u_{n+1}, then they are replaced by a "1"; otherwise, the neighbors are replaced by a "0." The process can be implemented by using

$$B_i = \begin{cases} 1 & if\ u_i \geq u_{n+1} \\ 0 & Otherwise, \end{cases} \tag{8}$$

where u_i's are neighbors of the center element u_{n+1} and $I = 1, 2, 3, \ldots, n, n+2, \ldots, 2n+1$. B_i is a row vector of length $2n$ containing the sequence of 0's and 1's. By considering the left most element in B_i as the most significant bit and the right most element as the least significant bit, the sequence in B_i is converted to a decimal digit by using the relation in Eq. (9).

$$L = B_1 \times 2^{2n-1} + B_2 \times 2^{2n-2} + \cdots + B_n \times 2^n + B_{n+2} \times 2^{n-1} + \cdots + B_{2n} \times 2^1 + B_{2n+1} \tag{9}$$

B_1 is multiplied by the highest power of 2, that is, 2^{2n-1} because it is the most significant bit. The decimal digit L is our required LBP code. The center element of the window is replaced by the computed LBP code L. Similarly, the LBP code for each window is computed, and the center elements are replaced with their corresponding code. In this way, each element of the forged audio U_{CM} is substituted by an LBP code. The range of LBP code depends on the number of bits in vector B. In the case of $2n$ neighbors (n on each side), the minimum code is 0 (when all $2n$ bits are zero) and the maximum code is 2^{2n-1} (when all $2n$ bits are one). The numbers of bins of a histogram are according to the number of LBP codes. Each bin represents the frequency of the corresponding LBP code.

The entire computation process of the LBP code is presented in Fig. 4. Suppose that the forged audio U_{CM} is provided by the following samples [0.7, 0.8, 0.5, 0.6, 0.1, 0.4, 0.7, 0.3, and 0.2]. The length of U_{CM} is $N = 9$ and is divided into windows of length $2n + 1 = 5$, where the number of neighbors on each side of the center elements is $n = 2$. The total number of the window of length 5 is $N - 2n = 5$ because

no zero padding is conducted on either end of the audio. The center element $n + 1 = 3$ is compared with each neighbor in every window. In case of the first window [0.7, 0.8, 0.5, 0.6, 0.1], the 3rd element is the center element and is equal to 0.5. The element 0.5 is compared with all of its neighbors [0.7, 0.8, 0.6, 0.1]. The neighbors 0.7, 0.8, and 0.6 are greater than the center element 0.5; therefore, they are replaced by "1." The neighbor 0.1 is smaller than the center element; hence, it is substituted by "0." Consequently, a row vector $B = [B_1, B_2, B_4, B_5] = [1\ 1\ 1\ 0]$ is obtained, which contains a sequence of 0's and 1's of length $2n = 4$. The left most element B_1 is the most significant bit. The most significant bit B_1 is multiplied with $2^{2n-1=3}$ to convert the four-bit binary number B into a decimal number. The remaining bits B_2, B_4, and B_5 are multiplied with $2^{2n-2=2}$, $2^{n-1=1}$, and 2^0, respectively. The sum of these terms provided an LBP code. The binary number contains four bits; therefore, the range for LBP code is from 0 $(0000)_2$ to $2^4 - 1 = 15$ $(1111)_2$. In this scenario, the number of LBP codes is 16; therefore, the histogram will contain 16 bins. Each bin describes the frequency of the corresponding LBP code. In other words, bin 10 represents the number of times LBP code 10 is repeated.

2) DETECTION OF COPY-MOVE FORGERY

The numbers of bins in the histograms depend on the total number of neighbors in a window. If the numbers of neighbors are $2n$, then the numbers of bins are 2^{2n} in a histogram. The size of histograms will be the same for all words because the size of the window is constant for all words of a forged audio. For the m number of words in a forged audio, m number of histograms is computed (one for each word). All histograms will be compared with each other to detect the forgery location and the word that has been copied and moved. A quality metric (QM) with two measures, mean square error (MSE), and energy ratio (ENR) is used to make comparisons between the histograms. MSE and ENR are computed by using the relation provided in Eqs. (10) and (11), respectively.

$$MSE = \frac{\sum_{g=1}^{2^{2n}} \left(H_i^g - H_j^g\right)^2}{2^{2n}}, \tag{10}$$

$$ENR = \frac{\sum_{g=1}^{2^{2n}} \left(H_i^g\right)^2}{\sum_{g=1}^{2^{2n}} \left(H_j^g\right)^2}, \tag{11}$$

where g stands for the number of bins in each histogram.

MSE determines the difference between two histograms by computing the squared sum of differences of corresponding bins in the two histograms H_i and H_j. The value of MSE closer to zero will indicate that histograms H_i and H_j are very similar to each other. Meanwhile, the large values indicate that the histograms are distinct. In case of copy-move forgery when the same word is present at locations i and j, MSE will provide a value equal to zero or very close to zero for the histograms H_i and H_j. The small values of MSE show that the words at locations i and j are the same. In this way, the words at locations i and j are determined to be copies of each other, and the audio under consideration is forged. In addition, ENR computes the energy of the histograms H_i and H_j by taking the sum of squared values of all bins and providing a ratio between energies of H_i and H_j. The value of ENR closer to 1 shows that the histograms H_i and H_j are the same.

For the m number of words, a matrix M of dimension $m \times m$ is obtained after comparing the histograms with each other. The first row $1 \times m$ of the matrix M contained the QM for the histogram of the first word with all remaining $m - 1$ histograms. Similarly, the QM for the second word is listed in the second row $2 \times m$ of the matrix M. At the diagonal of matrix M, all measures in QM will be zero. Diagonal elements 1×1, 2×2, 3×3, $\ldots$, $m \times m$ comprise values that are obtained by comparing the histogram of a word with itself. Therefore, we have to analyze the values of QM where i and j are different, that is $i \neq j$, to detect the copy-move forgery.

III. EXPERIMENTAL RESULTS FOR FORGERY DETECTION

Ninety different speakers of various nationalities are randomly taken from KSUD to generate copy-move forged audios and evaluate the performance of the proposed detection system. Recordings of all environments (soundproof room, office, and cafeteria) for each of the selected speaker are considered. Therefore, the total number of original audio recordings is 270 $(= 90 \times 3)$. These recordings contain 10 digits from zero to nine. The digits are considered because of their potential application to authenticating the secret code for remotely accessing the confidential data.

The sequence y to determine the locations are chaotically created by using initial condition $x_0 = 12$ and $y_0 = 9$ in Eqs. 4 and 5, respectively, to avoid human involvement in the generation of forged audio. Any value can be selected as an initial condition because we will scale this value later according to the requirement. Five recordings from each original audio with forgery at different locations are generated. For this purpose, 10 locations are required for each recording, one to determine the location to copy a digit and the other to move the digit. Therefore, the sequence y will contain at least 2700 $(= 270 \times 10)$ elements to generate 1350 forged audio recordings. Each audio has 10 locations. Therefore, a subsequence y_c of sequence y containing 10 numbers is scaled from one to ten using $u = 1$ and $v = 10$ in Eq. (6).

Each forged audio contains 10 digits, and a histogram for each digit is computed by following the process described in Section 2.2.1. The histograms are computed by considering $n = 2$, $n = 3$, and $n = 4$, where n represents the number of neighbors on each side of the center element in divided windows. The range of LBP codes for $n = 2$, 3, and 4 will be [0 15], [0 63], and [0 255], respectively. Therefore, the

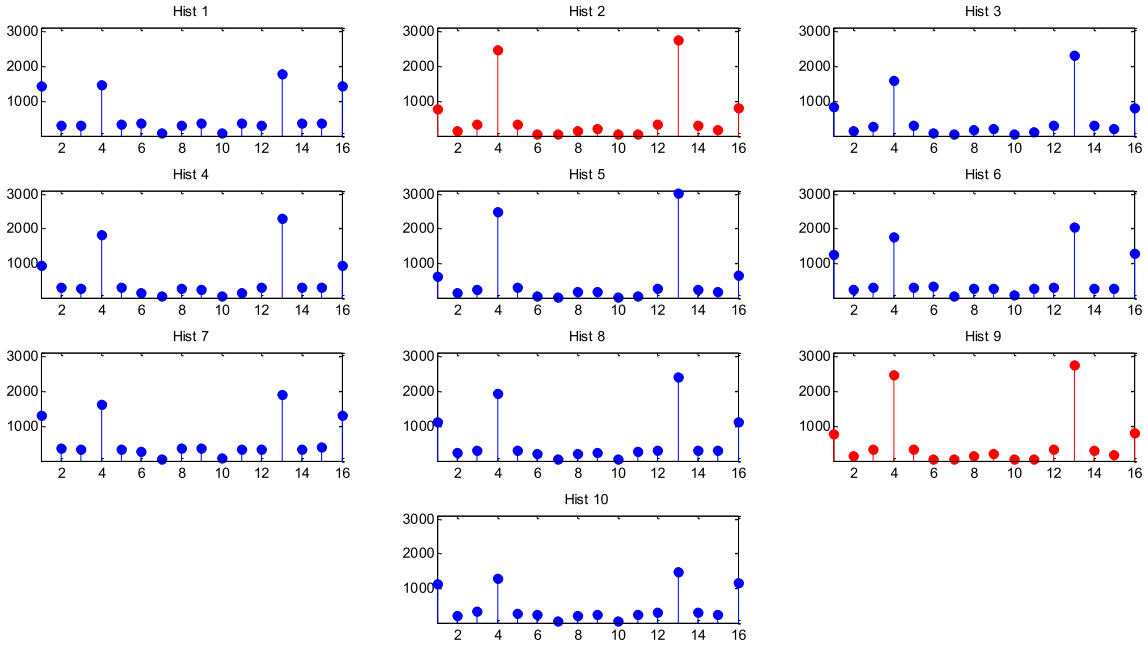

FIGURE 5. Computed histograms of digits from zero to nine in a copy-move forged audio for n = 2. Histogram Hist 1 is for digit 0, Hist 2 for digit 1 and so on.

TABLE 1. MSE for a forged audio when digit one is copied and moved to the place of digit eight.

Digits	0	1	2	3	4	5	6	7	8	9
0	0.0	1729.1	1130.2	982.6	2047.5	500.9	283.7	944.9	1728.3	628.6
1	1729.1	0.0	987.3	855.6	391.0	1244.4	1488.5	816.9	***0.0***	1802.4
2	1130.2	987.3	0.0	359.1	1201.1	774.4	932.2	613.5	986.4	998.3
3	982.6	855.6	359.1	0.0	1129.7	560.1	738.1	342.9	854.8	1009.0
4	2047.5	391.0	1201.1	1129.7	0.0	1565.3	1805.1	1115.8	391.2	2101.1
5	500.9	1244.4	774.4	560.1	1565.3	0.0	317.3	477.7	1243.7	765.5
6	283.7	1488.5	932.2	738.1	1805.1	317.3	0.0	724.5	1487.8	659.6
7	944.9	816.9	613.5	342.9	1115.8	477.7	724.5	0.0	816.0	1139.3
8	1728.3	***0.0***	986.4	854.8	391.2	1243.7	1487.8	816.0	0.0	1801.7
9	628.6	1802.4	998.3	1009.0	2101.1	765.5	659.6	1139.3	1801.7	0.0

number of bins in the histograms will be 16, 64, and 256 for $n = 2$, 3, and 4, respectively. The calculated LBP codes replace every sample of a digit, and these codes describe the characteristics of a digit. The computed histograms for all digits from zero to nine of a forged audio with $n = 2$ are depicted in Fig. 5.

All histograms are compared with each other, and measures in QM are calculated to detect the location of copy-move forgery. After comparison, a matrix M of dimension 10×10 is obtained. The calculated MSE for all histograms shown in Fig. 5 are listed in Table 1. The values for MSE in the first row are obtained by comparing the histogram of digit zero with all the remaining histograms. Similarly, the second row contains the MSE values of the histogram of digit one with the rest of the histograms.

The decision of the forgery is made by determining the index for the minimum value of MSE when $i \neq j$. Table 1 shows that MSE is equal to zero at row 1 (R1) and column 8 (C8) other than diagonal values, that is, $i \neq j$. This finding suggests that the digit at R1 and C8 are the same, with one of them as the original and the other one a copy. Therefore, it can be concluded that the audio is tampered by copy-move forgery, and the proposed system will highlight the original digit and its copy, as shown in Fig. 6. The other measure in QM, energy ratio, at R1 and C8, is equal to one. This ratio shows that both histograms have the same energy, which means that bins of both histograms are exactly equal, that is, representing the identical digits.

In the case of different copy-move locations in an audio recording, the MSE are presented in Tables 2 and 3. Table 2

TABLE 2. MSE for a forged audio when digit two is copied and moved to the place of five.

Digits	0	1	2	3	4	5	6	7	8	9
0	0.0	1729.1	1130.2	983.8	2047.5	1129.9	283.6	945.9	1391.2	628.6
1	1729.1	0.0	987.3	819.4	391.0	988.2	1488.7	816.1	697.9	1802.4
2	1130.2	987.3	0.0	437.3	1201.1	***0.0***	932.4	613.2	879.6	998.3
3	983.8	819.4	437.3	0.0	1101.0	437.7	734.1	276.7	668.8	1068.4
4	2047.5	391.0	1201.1	1101.0	0.0	1201.9	1805.2	1114.8	915.7	2101.1
5	1129.9	988.2	***0.0***	437.7	1201.9	0.0	932.3	613.6	880.1	997.8
6	283.6	1488.7	932.4	734.1	1805.2	932.3	0.0	725.5	1190.5	659.7
7	945.9	816.1	613.2	276.7	1114.8	613.6	725.5	0.0	570.0	1140.1
8	1391.2	697.9	879.6	668.8	915.7	880.1	1190.5	570.0	0.0	1606.9
9	628.6	1802.4	998.3	1068.4	2101.1	997.8	659.7	1140.1	1606.9	0.0

TABLE 3. MSE for a forged audio when digit three is copied and moved at the place of six.

Digits	0	1	2	3	4	5	6	7	8	9
0	0.0	1666.2	1130.3	941.8	1987.4	454.2	942.4	882.1	1311.7	734.2
1	1666.2	0.0	987.3	819.4	391.0	1244.4	818.9	816.1	697.9	1802.4
2	1130.3	987.3	0.0	437.3	1201.1	774.4	437.2	613.2	879.6	998.3
3	941.8	819.4	437.3	0.0	1101.0	550.5	***0.0***	276.7	668.8	1068.4
4	1987.4	391.0	1201.1	1101.0	0.0	1565.3	1100.3	1114.8	915.7	2101.1
5	454.2	1244.4	774.4	550.5	1565.3	0.0	551.1	478.8	974.5	765.5
6	942.4	818.9	437.2	***0.0***	1100.3	551.1	0.0	276.4	667.9	1069.2
7	882.1	816.1	613.2	276.7	1114.8	478.8	276.4	0.0	570.0	1140.1
8	1311.7	697.9	879.6	668.8	915.7	974.5	667.9	570.0	0.0	1606.9
9	734.2	1802.4	998.3	1068.4	2101.1	765.5	1069.2	1140.1	1606.9	0.0

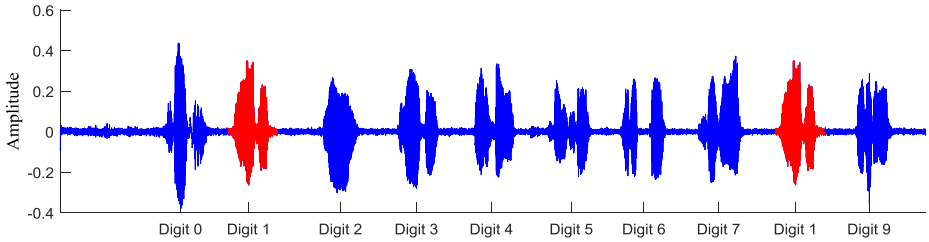

FIGURE 6. Original digit and its copy in a forged audio is highlighted by the proposed method.

provides the MSE values when the digit two is copied and moved at the place of five, whereas Table 3 provides the MSE values when the copy of digit three is moved at the place of six.

The proposed system is used for the localization of copy-move forgery in all generated forged recordings. All 1350 forged recordings are used to detect the copy-move forgery, and an accuracy of 96.59% is obtained. Although the language of the forged database is Arabic, the proposed system will still work for any language.

IV. ROBUSTNESS AND ANALYSIS OF THE PROPOSED METHOD

The proposed system should be able to detect the location of a copy-move forgery even in a noise added forged recording. The robustness of the system is tested by adding noise of different signal-to-noise ratios (SNR). Moreover, the comparisons of the proposed system with the contemporary methods are also presented in this section.

A. ROBUSTNESS AGAINST ATTACK OF NOISE

In case of copy-move forgery, an attacker can add random noise in an audio to hide the tampering. The random noise does not remain the same throughout the recording. This noise will be different at different places in a recording. Therefore, after adding the noise, the resultant forged audio will be changed. Random noise of various SNRs is added to the forged recordings to observe the robustness of the proposed system and then copy-move forgery is detected. In Fig. 7, a white Gaussian random noise of 30 dB is added to a forged signal.

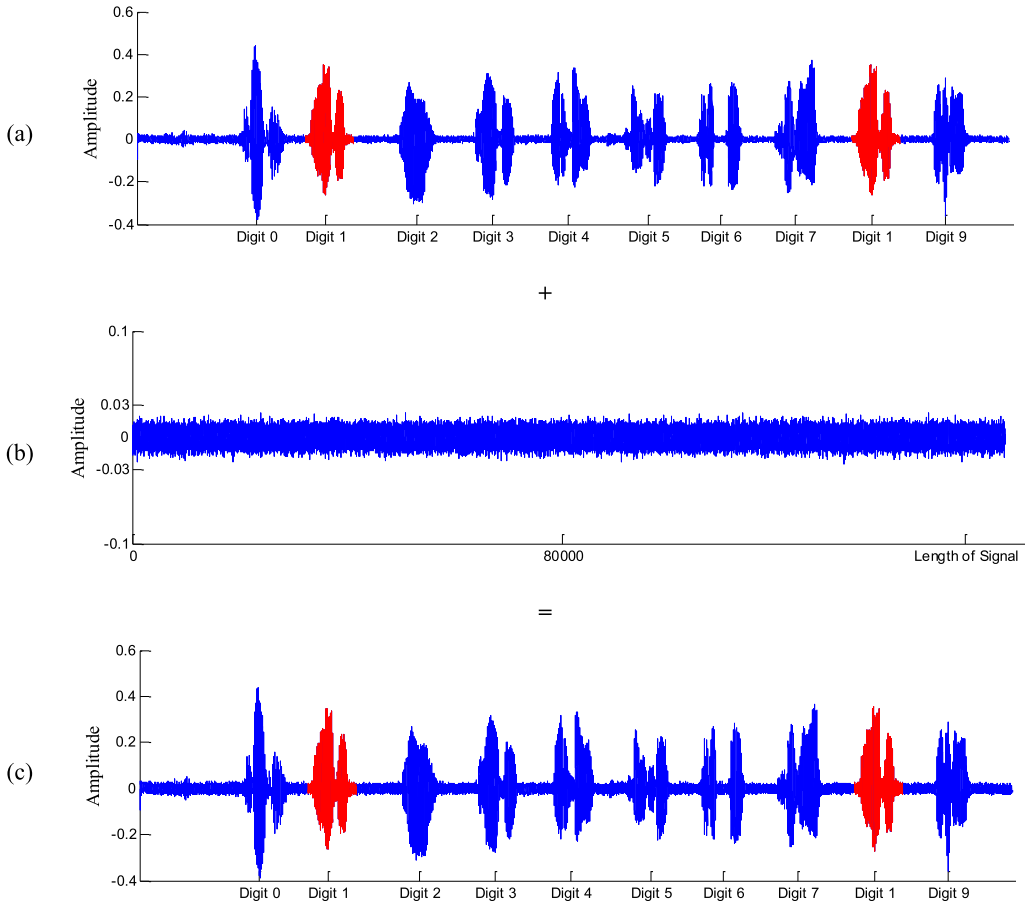

FIGURE 7. (a) Forged audio (b) White Gaussian random noise (c) Forged audio after adding noise (forged audio attacked with noise).

Detection of copy-move forgery may become apparent if the blind addition of noise in an audio either makes the entire signal suspicious or entirely distorts the signal. Therefore, noise is sophisticatedly added in the forged audio. Thus, no human can judge it by hearing. The power of the noise should be in accordance with the power of forged audio. The power of the forged audio F_A is computed as follows:

$$pow\,(F_A) = \frac{1}{N}\sum_{l=1}^{N}(F_A)^2. \tag{12}$$

Noise of SNR = 50 dB is added to all forged recordings and is increased by considering the SNR of 40 and 30 dB. Moreover, SNR of 20 dB can be significantly heard in the forged audio; therefore, SNR of 20 dB and below is not considered in the study. The power of the audio is converted into dB by using Eq. (13)

$$pow\,(F_A) = 10 \times Log_{10}\,[pow\,(F_A)]\ \text{dB}. \tag{13}$$

The power of noise is obtained by

$$pow\,(Noise) = pow\,(F_A) - SNR. \tag{14}$$

The performance of the proposed system is evaluated on noise added forged audio. The obtained accuracy of the system is 96.59%, which is similar to that of the forged audio without noise. No change in the accuracy shows that the addition of noise did not change the performance of the proposed system. This finding concludes that the proposed system is robust against noise and can detect copy-move forgery in case of noise attacks. The random noise changes the amplitude throughout the forged audio; therefore, the digit and its copy are different. The difference between their histograms is not exactly zero. However, the difference between histograms of a digit and its copy will be minimum. The histogram with a minimum distance determines the location of copy-move forgery.

B. ANALYSIS AND COMPARISON

This study has two important parts: the first part is the development of a copy-move forged audio database, and the second part described the proposed method for forgery detection and localization. Audio recordings of various speakers with different ethnic groups are selected to generate the forged database. For each speaker, the recording of three different

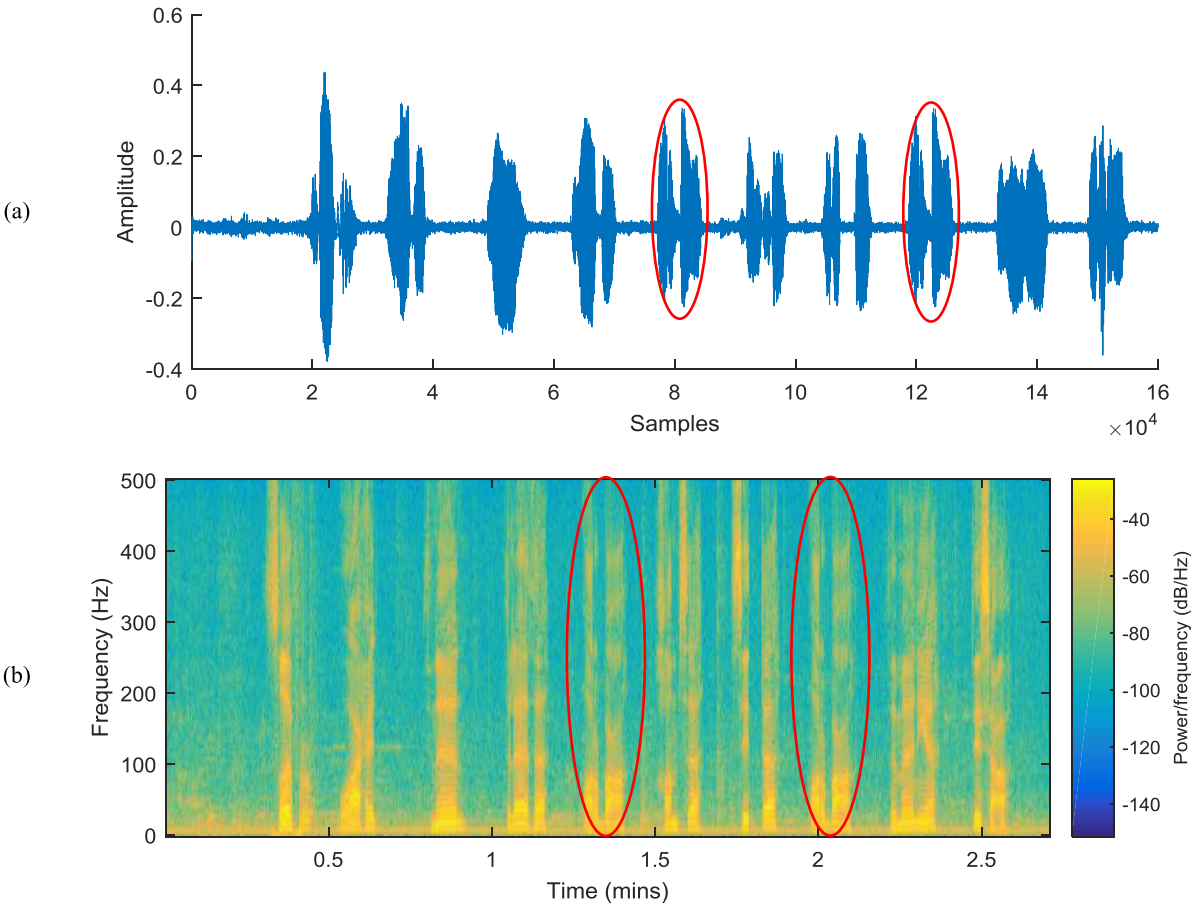

FIGURE 8. (a) A copy-move forged generated audio. (b) Spectrogram of the forged audio. No obvious clues are present in the spectrogram to detect the forgery.

environments and equipment is considered. By using these audios, the copy-move forged recordings are generated in such a way that an obvious trace exists to detect the forgery. Fig. 8 (a) depicts a forged recording in which digit four is copied and moved to the place of seven. The spectrogram of that audio is shown in Fig. 8 (b), in which the spectrogram shows that the generated forged sample does not have any irregularities or abrupt changes. Therefore, no one can judge by hearing and visualization that the audio is tampered.

The second part of the study presented a method for the localization of the copy-move forgery in audio recordings. The proposed method does not require watermark embedding in the original audio to determine the forgery. The method analyzes the contents of the audio to identify the copy-move forgery. The forgery is detected by finding the similarity between the words of audio recordings. The similar words (original word and its copy) are determined by the QM, which is obtained through the comparison of histograms. In the case of duplicate digits (a digit and its copy), MSE and ENR in QM will be close to zero and one, respectively. In addition, if a digit is repeating in a forged sample then each instance of the digit will have a different waveform [23]. However, waveforms of the digits will only be same in the case of copy-move forgery.

The accurate calculation of boundary points is one of the crucial components of the proposed method. Although the obtained accuracy is 96.59%, it can still be improved to 100% if it can be assured that the boundary points are accurately computed and that they do not contain any silence part. For forgery localization, the tuning of the different parameters is always essential [8], [9], [23]. For example, the authors of a previous study [23] used a threshold on the two parameters to make the decision regarding copy-move forgery, and the results depended on the adjusted thresholds. Similarly, the performance of the system for splicing forgery localization in [8] is sensitive to the sampling frequencies and requires tuning for different sampling frequencies.

Five different parameters are adjusted every time to tune the system. In addition, this method is very sensitive to noise, and the accuracy is decreased by 30% for noise added to the recordings. By contrast, our proposed method does not need any parameter adjustment. Moreover, the accuracy of the proposed system is not affected by the noise. Unlike [8], our proposed method is insensitive to sampling frequency

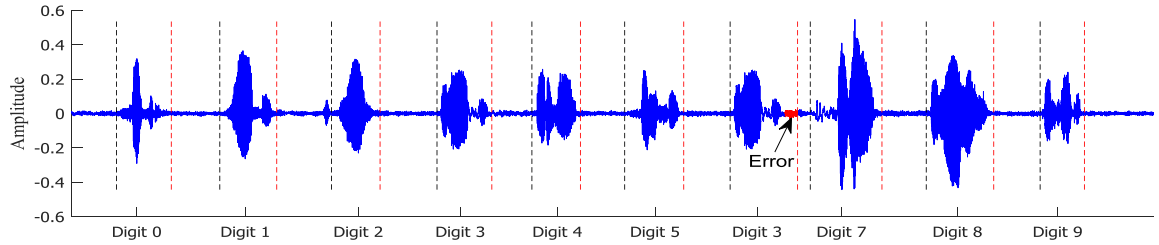

FIGURE 9. Error in detection of boundary points causes problem in forgery detection.

because it does not transform the audio recording into a frequency domain. The method proposed in [9] for splicing forgery localization is sensitive to the length of the analysis window. The accuracy of the method significantly varies because of the length of the window. However, the performance accuracy of our proposed method is not affected by the length of the analysis window and remains consistent for different windows, that is, $n = 2$, 3, and 4.

Nonetheless, in 3.41% of the forged recordings, copy-move forgery is incorrectly detected by the proposed method. These recordings are investigated, and boundary points are incorrectly determined. When boundary points are inaccurately calculated for the words in a recording, histograms of even similar words (original and its copy) will differ from one another. Therefore, detecting the copy-move forgery is impossible. A forged audio in which digit three is copied and moved to the place of six is shown in Fig. 9. In this forged audio, boundary points are inaccurately calculated.

The portion indicated by an arrow in Fig. 9 is not part of digit three. Therefore, the histograms are different, and MSE between digit three and its copy is not zero. The computation error of boundary points can be avoided if they could be manually double-checked to confirm that the detected boundary points do not contain silence.

V. CONCLUSION

In this paper, we have presented a new method for the detection and localization of copy-move forgery using the passive approach. The proposed method performs blind detection of forgery by inspecting the content of an audio recording. The method detects the words through the VAD module and computes the histograms by LBP application. By comparing the histograms, the method determines the similarity measures to make the decision regarding the location of copy-move forgery. An accuracy of 96.56% is obtained with the proposed method. The method does not need any threshold to make the decision, and no tuning of any parameter is required. By adding the noise of different signal-to-noise ratios, the performance of the system remains consistent. Moreover, attacking noise to hide the tampering of an audio does not affect the accuracy of the proposed method. During evaluation of the proposed method, a detection error of 3.41% is observed because of the inaccurate estimation of the boundary points. This error can be reduced to a minimum level in the future by improving the VAD module.

REFERENCES

[1] Z. Ali, M. Imran, and M. Alsulaiman, "An automatic digital audio authentication/forensics system," *IEEE Access*, vol. 5, pp. 2994–3007, 2017.

[2] S. Gupta, S. Cho, and C.-C. J. Kuo, "Current developments and future trends in audio authentication," *IEEE Multimedia Mag.*, vol. 19, no. 1, pp. 50–59, Jan. 2012.

[3] Audacity Team. (2016). *Audacity(R): Free Audio Editor and Recorder. Version 2.1.2*, accessed on Nov. 25, 2016. [Online]. Available: http://www.audacityteam.org/

[4] GoldWave Inc. (2016). *GoldWave: Digital Audio Editing Software. Version 6.24*, accessed on Nov. 25, 2016. [Online]. Available: https://www.goldwave.com/goldwave.php

[5] K. Asghar, Z. Habib, and M. Hussain, "Copy-move and splicing image forgery detection and localization techniques: A review," *Austral. J. Forensic Sci.*, vol. 49, no. 3, pp. 281–307, 2016.

[6] R. C. Maher, "Overview of Audio Forensics," in *Intelligent Multimedia Analysis for Security Applications*, H. T. Sencar, S. Velastin, N. Nikolaidis, and S. Lian, Eds. Berlin, Germany: Springer, 2010, pp. 127–144.

[7] K. Khaldi and A. Boudraa, "Audio watermarking via EMD," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 3, pp. 675–680, Mar. 2013.

[8] J. Chen, S. Xiang, H. Huang, and W. Liu, "Detecting and locating digital audio forgeries based on singularity analysis with wavelet packet," *Multimedia Tools Appl.*, vol. 75, no. 4, pp. 2303–2325, Feb. 2016.

[9] H. Zhao, Y. Chen, R. Wang, and H. Malik, "Audio splicing detection and localization using environmental signature," *Multimedia Tools Appl.*, vol. 76, no. 12, pp. 13897–13927, Jun. 2016.

[10] A. J. Cooper, "Detecting butt-spliced edits in forensic digital audio recordings," in *Proc. 39th Int. Conf., Audio Forensics, Pract. Challenges*, Jun. 2010, p. 1.

[11] X. Pan, X. Zhang, and S. Lyu, "Detecting splicing in digital audios using local noise level estimation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2012, pp. 1841–1844.

[12] G. Muhammad, Y. A. Alotaibi, M. Alsulaiman, and M. N. Huda, "Environment recognition using selected MPEG-7 audio features and mel-frequency cepstral coefficients," in *Proc. 5th Int. Conf. Digit. Telecommun.*, Jun. 2010, pp. 11–16.

[13] C. Kraetzer, A. Oermann, J. Dittmann, and A. Lang, "Digital audio forensics: A first practical evaluation on microphone and environment classification," presented at the 9th Workshop Multimedia Secur., Dallas, TX, USA, 2007.

[14] R. Buchholz, C. Kraetzer, and J. Dittmann, "Microphone classification using Fourier coefficients," in *Proc. 11th Int. Workshop Inf. Hiding*, Berlin, Germany, 2009, pp. 235–246.

[15] H. Q. Vu, S. Liu, X. Yang, Z. Li, and Y. Ren, "Identifying microphone from noisy recordings by using representative instance one class-classification approach," *J. Netw.*, vol. 7, no. 6, pp. 908–917, 2012.

[16] L. Cuccovillo, S. Mann, M. Tagliasacchi, and P. Aichroth, "Audio tampering detection via microphone classification," in *Proc. IEEE 15th Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2013, pp. 177–182.

[17] Z. Jinhua *et al.*, "Audio recorder forensic identification in 21 audio recorders," in *Proc. IEEE Int. Conf. Progr. Informat. Comput. (PIC)*, Dec. 2015, pp. 153–157.

[18] W. M. Campbell, K. J. Brady, J. P. Campbell, R. Granville, and D. A. Reynolds, "Understanding scores in forensic speaker recognition," in *Proc. IEEE Odyssey-Speaker Lang. Recognit. Workshop*, Jun. 2006, pp. 1–8.
[19] C. Champod and D. Meuwly, "The inference of identity in forensic speaker recognition," *Speech Commun.*, vol. 31, nos. 1–2, pp. 193–203, Jun. 2000.
[20] T. Thiruvaran, E. Ambikairajah, J. Epps, and E. Enzinger, "A comparison of single-stage and two-stage modelling approaches for automatic forensic speaker recognition," in *Proc. IEEE 8th Int. Conf. Ind. Inf. Syst.*, Dec. 2013, pp. 433–438.
[21] F. Guapo, P. Correia, D. Meuwly, and D. van der Vloed, "Empirical validation of likelihood ratio methods—A case study in forensic speaker recognition," in *Proc. 4th Int. Conf. Biometrics Forensics (IWBF)*, Mar. 2016, pp. 1–5.
[22] J. P. Campbell, W. Shen, W. M. Campbell, R. Schwartz, J. F. Bonastre, and D. Matrouf, "Forensic speaker recognition," *IEEE Signal Process. Mag.*, vol. 26, no. 2, pp. 95–103, Mar. 2009.
[23] Q. Yan, R. Yang, and J. Huang, "Copy-move detection of audio recording with pitch similarity," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 1782–1786.
[24] S. A. Zahorian and H. Hu, "A spectral/temporal method for robust fundamental frequency tracking," *J. Acoust. Soc. Amer.*, vol. 123, no. 6, pp. 4559–4571, 2008.
[25] M. Alsulaiman, G. Muhammad, B. Abdelkader, A. Mahmood, and Z. Ali, *King Saud University Arabic Speech Database, Hard Drive*, catalog LDC2014S02, Linguistic Data Consortium, Philadelphia, PA, USA, 2014.
[26] M. M. Alsulaiman, G. Muhammad, M. A. Bencherif, A. Mahmood, and Z. Ali, "KSU rich arabic speech database," *Information*, vol. 16, no. 6, pp. 4231–4253, 2013.
[27] M. Alsulaiman, Z. Ali, G. Muhammed, M. Bencherif, and A. Mahmood, "KSU speech database: Text selection, recording and verification," in *Proc. Eur. Modelling Symp. (EMS)*, Nov. 2013, pp. 237–242.
[28] I.-C. Yoo, H. Lim, and D. Yook, "Formant-based robust voice activity detection," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 12, pp. 2238–2245, Dec. 2015.
[29] G. Aneeja and B. Yegnanarayana, "Single frequency filtering approach for discriminating speech and nonspeech," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 4, pp. 705–717, Apr. 2015.
[30] R. L. Devaney, "Fractal patterns arising in chaotic dynamical systems," in *The Science of Fractal Images*, H.-O. Peitgen D. Saupe, Eds. New York, NY, USA: Springer, 1988, pp. 137–168.

MUHAMMAD IMRAN has been an Assistant Professor with the College of Computer and Information Sciences, King Saud University, Riyady, Saudi Arabia, since 2011. He is currently a Visiting Scientist with Iowa State University, Ames, IA, USA. He has authored a number of high-quality research papers in refereed international conferences and journals. His research is financially supported by several grants. His research interest includes mobile ad hoc and sensor networks, WBANs, M2M, Internet of Things, SDN, fault tolerant computing, and security and privacy. Recently, European Alliance for Innovation (EAI) appointed him as Co-Editor in Chief for *EAI Transactions on Pervasive Health and Technology*. He also serves as an Associate Editor of the IEEE Access, the *IEEE Communications Magazine*, *Wireless Communication*, the *Mobile Computing Journal*, the *Ad Hoc and Sensor Wireless Networks Journal*, *IET Wireless Sensor Systems*, the *International Journal of Autonomous and Adaptive Communication Systems*, and the *International Journal of Information Technology and Electrical Engineering*. He served/serving as a Guest Editor of the *IEEE Communications Magazine* (SCIE), *Computer Networks* (SCIE, Elsevier), *MDPI Sensors* (SCIE), *International Journal of Distributed Sensor Networks* (SCIE, Hindawi), *Journal of Internet Technology* (SCIE), and *International Journal of Autonomous and Adaptive Communications Systems*. He has been involved in more than 50 conferences and workshops in various capacities, such as a chair, a co-chair, and a Technical Program Committee Member. These include the IEEE ICC, Globecom, AINA, LCN, IWCMC, IFIP WWIC, and BWCCA. He has received number of awards such as Asia–Pacific Advanced Network Fellowship.

ZULFIQAR ALI received the M.S. degree in computational mathematics from the University of the Punjab, Lahore, and the M.S. degree in computer science from the University of Engineering and Technology, Lahore, with the specialization in system engineering. Since 2010, he has been a Full-Time Researcher with the Digital Speech Processing Group, Department of Computer Engineering, King Saud University, Saudi Arabia. He is also a member of the Center for Intelligent Signal and Imaging Research, Universiti Teknologi PETRONAS, Malaysia. His research interests include speech and language processing, medical signal processing, privacy and security in healthcare, multimedia forensics, and computer-aided pronunciation training systems.

SHEIKH TAHIR BAKHSH received the Ph.D. degree in computer and information sciences from Universiti Teknologi PETRONAS, Malaysia, in 2012. He joined the Faculty of Computing and Information Technology, King Abdulaziz University, Saudi Arabia, as an Assistant Professor, in 2013. In the recent, he has completed the LTE HICI Project in collaboration with Stanford University, USA. He has also directed graduate and undergraduate projects. He has authored over 25 journal articles and referred conference papers in these areas. His areas of research interests include bluetooth network, wireless sensor network, mobile ad hoc network, and computer networks, wireless network protocol designs optimizing the performance of networks. Recently, he has been involved in projects related MAC protocol design for tele-monitoring. He received the Gold Medal by the Rector COMSATS Institute of Information Technology, Abbottabad, Pakistan, for securing the First position in MCS in 2006.

SHEERAZ AKRAM received the B.S. degree in computer science from the International Islamic University, Islamabad, Pakistan, in 2004, the M.S. degree in computer science from the Lahore University of Management Sciences, Lahore, in 2006, and the Ph.D. degree from the National University of Science and Technology, Islamabad, in 2017. He is currently an Assistant Professor with the Department of Software Engineering, Foundation University, Islamabad. He has been involved in teaching and research activities in various prestigious institutes/universities of Pakistan since 2004. His area of research is multimedia, medical signal processing, and forensics.