International Journal of Digital Crime and Forensics
Volume 11 • Issue 2 • April-June 2019

Fast and Effective Copy-Move Detection of Digital Audio Based on Auto Segment

Xinchao Huang, School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China

Zihan Liu, School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China

Wei Lu, School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China

Hongmei Liu, School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China

Shijun Xiang, College of Information Science and Technology, Jinan University, Guangzhou, China.

ABSTRACT

Detecting digital audio forgeries is a significant research focus in the field of audio forensics. In this article, the authors focus on a special form of digital audio forgery—copy-move—and propose a fast and effective method to detect doctored audios. First, the article segments the input audio data into syllables by voice activity detection and syllable detection. Second, the authors select the points in the frequency domain as feature by applying discrete Fourier transform (DFT) to each audio segment. Furthermore, this article sorts every segment according to the features and gets a sorted list of audio segments. In the end, the article merely compares one segment with some adjacent segments in the sorted list so that the time complexity is decreased. After comparisons with other state of the art methods, the results show that the proposed method can identify the authentication of the input audio and locate the forged position fast and effectively.

KEYWORDS

Auto Segment, Copy-Move Detection, DFT, Digital Audio Forensics

1. INTRODUCTION

With the continuous development of science, digital multimedia, especially digital audio, is widely used nowadays. Because digital audio is easy to be transmitted and stored, it makes our daily life more colorful. However, as is well-known that everything is a double-edged sword, digital audio can also cause harm to the society in that it is easy to be edited, or in other words, vulnerable. As a result, the authentication of digital audio is significant since it might play an important role like a piece of crucial evidence in forensics and court. What even worse is that some types of digital audio forgeries such as copy-move forgery are imperceptible, and it's difficult to be detected. Copy-move forgery of digital audio could be done as follows: copy some words from an original audio and paste the words to other positions of the same audio. It can be easily realized by using the audio editing application such as Adobe Audition CC and people can hardly detect the copy-move forgery through ears because of the copied segment derived from the same audio. In addition, some post-processing may be adopted to the copied segment for making the forgery harder to be detected. Therefore, copy-move forgery detection of digital audio has become an urgent issue in the area of audio forensics.

At present, some advanced technologies like digital watermarking and digital signature are used to protect the integrity of digital audio effectively. Such kind of technology is called active forensic

DOI: 10.4018/IJDCF.2019040104

technique. Many excellent audio watermarking algorithms (Bassia, Pitas & Nikolaidis, 2001; Wang & Zhao, 2006; Wu, Su & Kuo, 2000; Li, Xue & Lu, 2006; Xiang & Huang, 2007) have been proposed. However, the biggest limitation is that most of recording devices don't have the function to insert watermark or signature into digital audio data now. For this reason, another kind of technology, which is called passive forensic technique, is arousing attention in audio forensics nowadays. Passive forensic technique can just use the audio without adding any digital watermarking or signature for verifying the authenticity and integrity of audio, and our method for copy-move detection of digital audio is based on passive forensic technique.

There are many research achievements in the area of audio forensics. Farid (Farid, 1999) put forward to an assumption that in the frequency domain a natural signal has weak higher-order statistical correlations, and proposed a new scheme that use polyspectral analysis technique to detect the forgery. Cooper (Copper, 2010) analyzed the cross-correlation between the signal and second-order difference, and proposed a method that can detect the "butt-splicing'' in tempered audio. Alessandro (D'Alessandro & Shi, 2009) used frequency spectrum analysis to detect MP3 bit rate quality. Grigoras (Grigoras, 2005) proposed a new method that use ENF (electric network frequency) as feature for verifying the authenticity of audio. Maarten et al. (Huijbregtse & Geradts, 2009) improved Grigoras's method. They found that there are certain requirements about file length in Grigoras's method, only when file length reaches to a certain value could a precise result be gotten. So, Maarten et al. did some pre-processings for audio data at first, then calculated correlation coefficient and made improved algorithm effective for short-time audio files. Kraetzer et al. (Kraetzer, Oermann, Dittmann & Lang, 2007) detected the forgery by classifying the audio using statistical features of digital audio consisting of time domain-based features and mel-cepstral domain-based features, the method detects forgery by checking whether every audio frames were recorded under same circumstance or same equipment. Yang et al. (Yang, Qu & Huang, 2008; Yang, Qu & Huang, 2012) used the inconsistency of frame offset to detect the audio forgery in MP3 files. Chen et al. (Chen, Xiang, Liu & Huang, 2013) analyzed high-order singularity of wavelet coefficients and proposed an audio splicing detection model. Pan et al. (Pan, Zhang & Lyu, 2012) came up with another approach for audio splicing detection. They used local noise level estimation to detect the splicing digital audio.

As to content-based forgery detection in digital audio, Gupta et al. (Gupta, Boulianne & Cardinal, 2010) proposed a fingerprinting method that detect the copy by calculating the score between the query frame and the test frame. Another robust fingerprinting system was proposed by Ouali et al. (Ouali, Dumouchel & Gupta, 2015), they got the spectrogram of the digital audio first, then encoded the positions of salient regions of binary images which derived from the spectrogram as fingerprints. However, both of them focus more on content-based forgery and are useful for audio retrieval and monitoring of ad campaigns. Up to now, there are few works on the detection of copy-move in audio. Xiao et al. (Xiao, Jia, Fu, Huang, Li & Shi, 2014) proposed a method that detect the forgery by calculating the similarity between different segments. Another similar idea was proposed by Yan et al. (Yan, Yang & Huang, 2015) which based on pitch similarity. The method extracted the pitch of every syllable and calculated the similarities of these pitch sequences. However, Xiao's method just segmented the audio with fixed length and Yan's method didn't definitely indicate how to segment the audio.

There are two main evaluation criteria about a good copy-move detection method: accuracy and detection time. The step of audio segmentation determines the accuracy of detection to a great degree. Besides, the detection time is also an important judge. If a method can detect the copy-move forgery precisely but consumes large amounts of time, it is not so useful in practical application. Some state-of-the-art methods compare all the audio segments one by one, which is extremely time-consuming when the number of audio segments is large.

In this paper, we propose a novel method that can detect copy-move forgery of digital audio fast and effectively. The contributions of our method are as follows. Differ from other method that divide the audio in fixed length, which is inexact, we divide the audio by auto segment. This is based on the content of audio and the outlaws often tend to change the meaning or content of digital audio when tampering. The auto segment is used for deciding the locations and lengths of the segments that are divided from the audio. So our method is stricter and more precise than other methods in audio segmentation. Moreover, an extra sorting step is added. It can decrease the time complexity of similarity comparison step from $\mathrm{O}\left(n^2\right)$ to $\mathrm{O}\left(n\log n\right)$. In a word, the proposed method can meet the criteria well and solve the problem of digital audio copy-move detection precisely and fast.

This paper's frame is as follows. Section 2 describes the proposed method's framework and the whole approach in detail. In section 3, experiments are conducted to evaluate the effectiveness of proposed method. In the end, the conclusions are drawn in Section 4.

2. PROPOSED APPROACH

In this paper, a novel method is proposed to detect digital audio copy-move forgery fast and effectively. The input audio data is segmented by auto segment, which including voice activity detection and syllable detection according to the characteristics of human voice. It makes the detection method more suitable for the forgery in the audio with human voice. Different features are tested in this paper, and we extract DFT feature from the segments that segmented from audio. For decreasing the time complexity of the proposed method, we also add a sorting step before the similarity comparison. Different from the other methods (Xiao, Jia, Fu, Huang, Li & Shi, 2014; Yan, Yang & Huang, 2015), whose similarity comparison are calculated between every segments, and the time complexity is $\mathrm{O}\left(n^2\right)$. The proposed method is improved by two-step strategy, which reduces the time complexity of the two steps to $\mathrm{O}\left(n\log n+n\right)$ and approximately equal to $\mathrm{O}\left(n\log n\right)$.

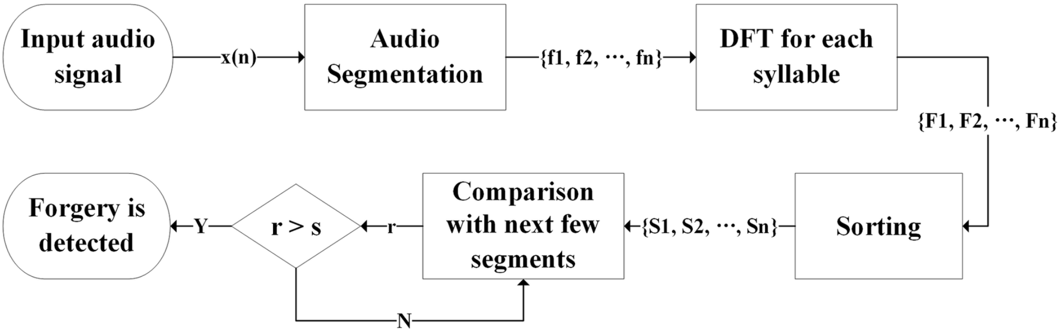
This section is organized as follow. First, Section 2.1 presents the whole framework of the proposed method. Second, in the following section, the approach of the proposed method is described in detail. Section 2.2 shows the splitting strategy of the proposed method. Section 2.3 presents the two different features that are selected to extract feature from the audio. Section 2.4 explains the sorting steps that added to reduce the time complexity of the next step. Section 2.5 shows the way of evaluating the similarity between audio segments.

2.1. Framework

The framework of the proposed method can be separated into four stages: audio segmentation, feature extraction, sorting, and similarity comparison. Audio segmentation is to segment the audio into syllables, it is a necessary preprocessing step and has an influence on the accuracy of detection. Feature extraction is to extract useful features of each syllable, which is used for similarity comparison. The feature is extracted by DFT. Sorting is like a boost for copy-move detection, it can make the time complexity of the last step much lower. Similarity comparison is the last step and it determines whether the audio is tampered directly. Choosing an appropriate criterion of similarity is essential and we select Pearson Correlation Coefficient (PCCs). The Figure 1 shows the whole framework of the proposed method.

As shown in Figure 1, the input digital audio signal $x\left(n\right)$ is divided into segments and $\left\{f_1,f_2,\ldots,f_n\right\}$ is the set of segments, f_i represents the i-th syllable segment. The list $\left\{F_1,F_2,\ldots,F_n\right\}$ represents the features of each segment, and the features are extracted by DFT. F_i contains the i-th syllable's feature. After sorting the segments on the basis of the characteristic value v_i of F_i, the sorted list $\left\{S_1,S_2,\ldots,S_n\right\}$ of audio segments is got. s is the threshold we set, r is the similarity between two features.

Figure 1. Framework of the proposed method



2.2. Audio Segmentation

Audio segmentation is the first step and also the essential step of the detection process. A precise segmentation can increase the accuracy of detection largely. The existing methods all split the audio in time domain but in different ways. For instance, Xiao's method (Xiao, Jia, Fu, Huang, Li & Shi, 2014) divided the audio file with a fixed time span T . Though different value of T was chosen according to different situations, it couldn't guarantee that the tampered regions wouldn't be damaged when dividing the audio file, because we know nothing about the tampered region like its length and the position in the audio before detection. As a result, we must divide the audio file dynamically.

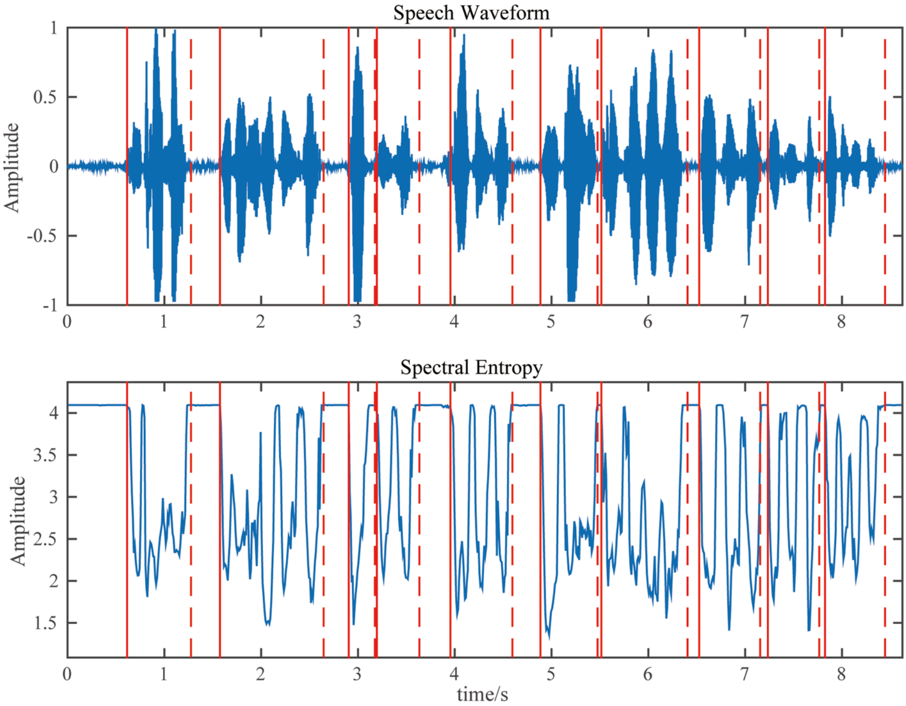
Consider the fact that if someone wants to tamper the digital audio in the form of copy-move, he will certainly copy one word or several words to other locations of the original audio and change the meaning of the audio to some extent. If we can split the audio word by word and compare the word segment with other word segments, then the problem can be solved more easily. However, splitting the audio word by word is difficult because the time length of each word is various. Even the same person speaks the same word at different time, the time length of the word is different. Nevertheless, a word consist of several syllables, we can subdivide the word into syllables and use syllables as the basic units for the next steps. It won't influence the accuracy of detection because we may detect the similarity of consecutive syllables and each syllable must be a part of one word.

In this step, we use two substeps to divide the input audio into syllables. At first, voice activity detection is used to eliminate the distraction of silence and divide the audio into several voiced sections. Then syllable detection is applied to subdivide each voiced section into several syllables.

2.2.1. Voice Activity Detection

In copy-move detection of digital audio, the step of voice activity detection aims at eliminating the influence of silence and short-time noise, thus improving the accuracy of the whole process. We use spectrum entropy as feature to do voice activity detection. Spectral entropy describes the complexity of a system, and it is very useful in distinguishing the speech segments in a continuously recorded utterance from the non-speech parts, especially under sophisticated noisy environments (Shen, Hung & Lee, 1998). Spectral entropy is defined as follows: assume input audio signal in time domain is $x\left(n\right)$, after windowing and framing, the i -th frame is $x_i\left(m\right)$. The fast Fourier transform (FFT) is used to get the spectrum form frames. For k -th frequency component f_k, the spectral energy $Y_i\left(k\right)$ is obtained. The probability density function for the spectrum can thus be estimated by normalization overall frequency components:

Figure 2. Result of voice activity detection. Red solid lines represent the origin of each voiced section, and red dash lines represent the destination.



$$p_i\left(k\right) = \frac{Y_i\left(k\right)}{\sum_{t=1}^{N/2} Y_i\left(l\right)} \tag{1}$$

where $p_i\left(k\right)$ is the corresponding probability density, and N is the total number of frequency components in FFT. Then the corresponding spectral entropy for each frame is defined as:

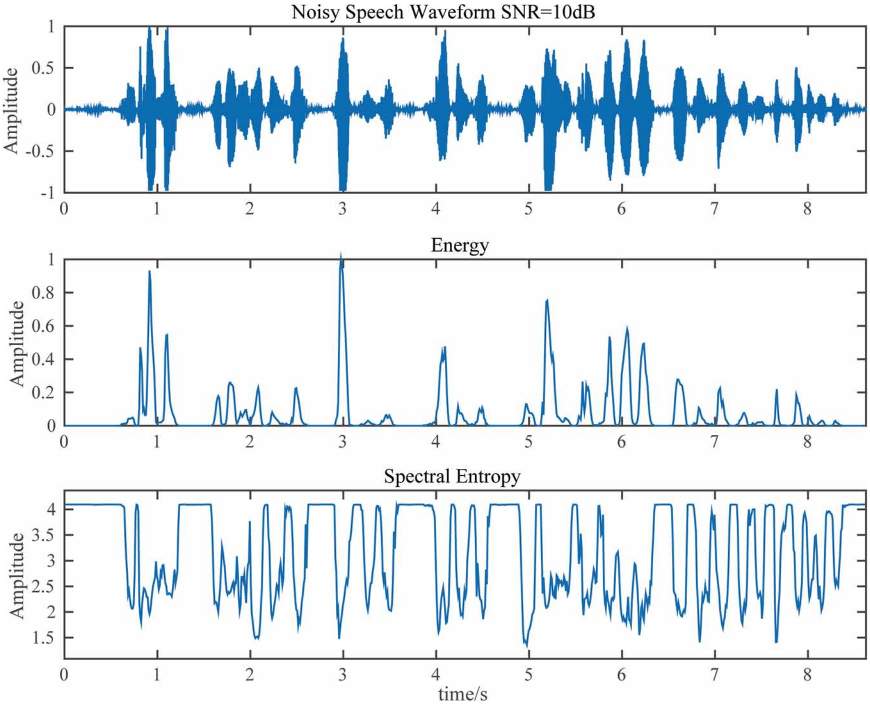
$$H_i = -\sum_{k=1}^{N/2} p_i\left(k\right) \log p_i\left(k\right) \tag{2}$$

After obtaining the spectral entropy values for each frame, the double-threshold method is adopted for voice activity detection. After voice activity detection, several voiced sections are got. Figure 2 shows the result of a sample audio through voice activity detection. The result shows that the voice activity detection is relatively accurate.

2.2.2. Syllable Detection

The syllable has proven to be an important concept in theoretical description of spoken language (Pfitzinger, Burger & Heid, 1996). For each voiced section, we use the ratio of energy to spectral entropy as feature to do syllable detection. Figure 3 shows the energy and spectral entropy of a noisy speech signal. From the figure, the energy curve of voiced section bulges upwardly, while the entropy curve is concave downwardly. It shows that in voiced section, the value of energy is large while the

Figure 3. Illustration for energy and spectral entropy waveform of a noisy speech signal



value of spectral entropy is small; in noise section, the situation is opposite. So the ratio of energy to spectral entropy can distinguish the voiced sections from noise sections better.

Assume that the input audio signal in time domain is $x\left(n\right)$, after windowing and framing, the i-th frame is $x_i\left(m\right)$, and the length of frame is N. The energy of each frame is defined as:

$$AMP_i = \sum_{m=1}^{N} x_i^2\left(m\right) \tag{3}$$

Here introduces improved energy calculation:

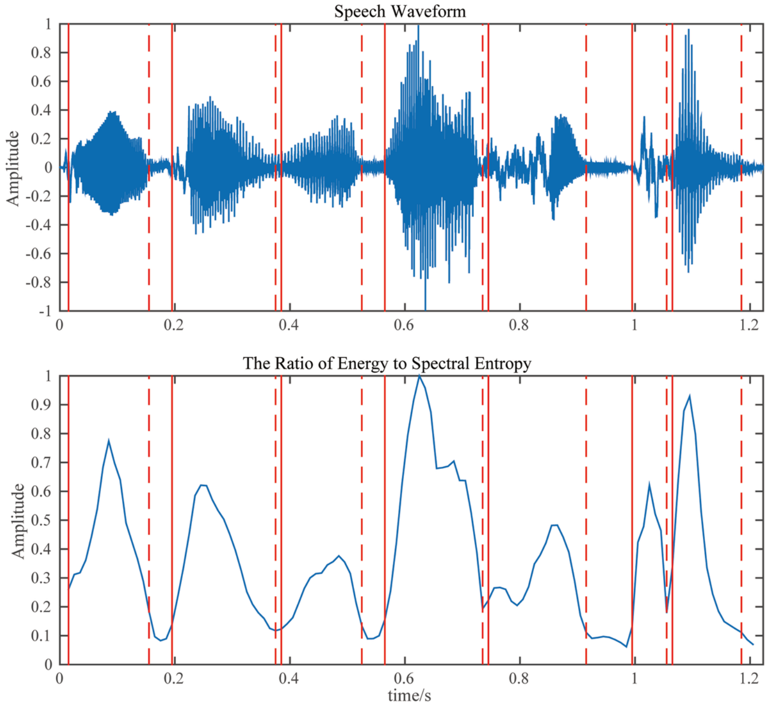
$$EL_i = \log_{10}\left(1 + AMP_i / a\right) \tag{4}$$

where AMP_i is the energy of each frame according to Equation (3); a is a constant. Because of the existence of a, if a takes a larger value, when the amplitude of AMP_i varies greatly, it will be eased in EL_i and help to distinguish the noise and the human voice when segment syllables.

The ratio of energy to spectral entropy is defined as:

$$EEF_i = \sqrt{1 + \left|EL_i / H_i\right|} \tag{5}$$

Figure 4. Result of syllable detection. Red solid lines represent the origin of each syllable, and red dash lines represent the destination.



where the energy is represented by EL_i according to Equation (4); the spectral entropy is represented by H_i according to Equation (2).

Syllable detection is similar to voice activity detection, but we only use one threshold *T* to segment syllables and the threshold *T* is stricter than the thresholds in voice activity detection. We find the part as a syllable whose the value of the ratio of energy to spectral is larger than *T* in each voiced section, and only keep the syllable whose time span is longer than a preset value *t*. Figure 4 shows the result of a voiced section through voice activity detection. The result shows that the syllable detection is relatively accurate.

2.3. Feature Extraction

Feature extraction is a crucial step in the copy-move detection. The extracted features must have the characteristic of high distinction if we want to distinguish the duplicated segments and normal segments more accurately. As is known to all, human voice can be decomposed into different frequency components, different person speak words in different way, the same person speak the same word at different time, the frequency compositions of the voice are different. However, the duplicated segment is very similar to the original one and the frequency compositions between original and duplicated segment are also similar. In our method, two different frequency domain features are selected: Discrete Fourier Transform (DFT) and Mel-Frequency Cepstral Coefficients (MFCCs).

DFT is an important way of signal processing. It transforms original input audio signal to frequency domain. A digital audio signal can be denoted with duration *T* seconds. If the sample rate is r_s Hz, we can use a 1-D vector $x\left(n\right)=\left[x_1,x_2,\ldots,x_{N-1}\right]$ with length N to represent the sampling points sequence, and $N=r_s{*}T$. Then the DFT of the input audio signal can be calculated as follow:

$$X\left(k\right) = DFT\left[x\left(n\right)\right] = \sum_{n=0}^{N-1} x\left(n\right) e^{-j\frac{2\pi}{N}kn} \tag{6}$$

where $k = 0,1,2,\ldots,N-1$.

MFCCs are commonly used as features in voice recognition systems. The MFCCs can be obtained from audio as follow:

1. Divide audio signal into frames;

Divided audio frames are transformed to frequency domain by using DFT, then frequency signal $P_i\left(f\right)$ is got and the short-term power spectrum $P_i\left(\omega\right)$ is calculated as:

$$P_i\left(\omega\right) = \left|P_i\left(f\right)\right|^2 \tag{7}$$

where i means i-th frame.

3. For power spectrum, we apply the mel filterbank to it, and the energies in each filter are summed. In the Equation (8), the given frequency f in Hz is converted to Mel scale:

$$M\left(f\right) = 2595\log_{10}\left(1 + f / 700\right) \tag{8}$$

Then calculate the filterbank energy outputs as follow:

$$\theta\left(M_k\right) = \sum_{k=1}^{k} \left|P_i\left(f\right)\right|^2 H_m\left(k\right) \tag{9}$$

where k means k-th filter, $k = 1,2,\ldots,K$, and K represents the number of filters. Besides, $H_m\left(k\right)$ represents K mel filterbanks.

4. Obtain the logarithm of filterbank energies: $X\left(k\right) = \ln\left(\theta\left(M_k\right)\right)$;

Calculate the DCT of logarithm of filterbank energies. The conversion is as follow:

$$MFCC\left(n\right) = \sum_{K=1}^{K} X_k \cos\left[n\left(k - 0.5\right)\frac{\pi}{K}\right] \tag{10}$$

where $1 \le n \le K / 2$. The 2-13 coefficients in Equation (10) are often keep as MFCCs, and discard the rest coefficients.

It is hard to determine which feature is better for similarity comparison. So, some experiments are made to select the best feature in Section 3.2. Finally, we found that DFT is more suitable for comparing audio segments.

2.4. Sorting

Generally, one segment is compared with all the other segments in the aspect of feature to find the similar segment. Obviously, the duplicated segment is copied from other segment. Even if the segments were processed by some operations, such as adding noise, the duplicated segments are still very similar to the original segments. Other normal segments are different with each other. So, it is completely redundant to compare some segments. So, we just need to compare a segment with some suspicious segments. In addition, the main frequency components of human voice concentrate on the low frequency, so we add a sorting step based on the low frequency part. And this step constructs a list of the feature of segments, which is sorted for comparing. In the sorted list, the similarity among the neighboring segments are calculated, and the duplicated segment could be detected fast by this step. For improving the accuracy of detection, we use first ^{2} points in the sequence for sorting the feature. The sum of the ^{2} points is regard as representation of the feature sequence and as the sorting elements. If we extract a feature sequence $X\left(k\right)=\left\{X_1,X_2,\ldots,X_{k-1}\right\}$ of an audio segment with k -length. The characteristic value v can be obtained as:

$$v=\sum_{k=0}^{\beta-1}X\left(k\right) \tag{11}$$

Then sorting is based on the characteristic values. As to the selection of sort algorithm, compared to other sort algorithms, we select quicksort in the proposed method because of its high efficiency.

2.5. Similarity Computation

In our method, we use Pearson Correlation Coefficient (PCCs) to decide whether two segments are similar or not. PCCs is the covariance of the two variables divided by the product of their standard deviations. The definition of Pearson Correlation Coefficient is as follows.

Define two feature sequence $\left\{x_1,x_2,\ldots,x_n\right\}$ and $\left\{y_1,y_2,\ldots,y_n\right\}$ which all contain n values, the PCCs of the two sequences can be obtained as:

$$r=\frac{\sum_{i=1}^{n}\left(x_i-\overline{x}\right)\left(y_i-\overline{y}\right)}{\sqrt{\sum_{i=1}^{n}\left(x_i-\overline{x}\right)^2}\sqrt{\sum_{i=1}^{n}\left(y_i-\overline{y}\right)^2}} \tag{12}$$

where $\overline{x}=\frac{1}{n}\sum_{i=1}^{n}x_i$; and so is the same of $\overline{y}$. The value of PCCs will get close to 1 while the comparing two feature are more similar and the range of it is $\left[-1,1\right]$.

If we obtain m feature sequences $\left\{F_0,F_1,\ldots,F_{m-1}\right\}$ which is extracted from a digital audio according to the segments. Then we can get the sorted list $\left\{S_0,S_1,\ldots,S_{m-1}\right\}$ after the sorting step. For each $S_i\left(0\le i\le m-\alpha-1\right)$, compare it with next ± sequences; for the rest of $S_i\left(m-\alpha\le i\le m-2\right)$, compare it with the rest sequences up to S_{m-1}. When comparing two sequences, if the value of PCCs is larger than the threshold, we consider that the copy-move forgery of digital audio exists. It is obvious that the time complexity of similarity computation is $\mathrm{O}\left(n\right)$. The time complexity of last two steps is $\mathrm{O}\left(n\log n+n\right)$. ($\mathrm{O}\left(n\log n\right)$ is the average time complexity of sort step), which is approximate to $\mathrm{O}\left(n\log n\right)$ and less than $\mathrm{O}\left(n^2\right)$.

Table 1. PCCs of repeated segments under different feature

PCCs	Over 0.9	0.8-0.9	0.7-0.8	0.6-0.7
DFT	0.03%	4.41%	38.06%	40.07%
MFCCs	77.27%	22.73%	0%	0%

3. EXPERIMENTAL RESULTS

In the following subsection, first, we introduce the audio dataset that prepared for the experiments in Section 3.1. Next, some experiments are made for determining the effective feature for detection in Section 3.2. Then the selection of threshold and parameter settings are discussed in Section 3.3, and the evaluation of our method is made in Section 3.4. In the end, the proposed method is compared with other state-of-art methods in Section 3.5.

3.1. Dataset

In the Dataset, we prepare 1000 audios and their doctored version with copy-move forgery for the experiment. All the audio is in the format WAV and the sampling rate is $16k$ Hz. The length of each audio is 3 to 4 minutes and the length of copied segments is between 0.2 and 0.6 seconds. Then we add noise and do filtering to the 1000 doctored audios respectively for post-processing. Total 3000 doctored audios are used in the experiments. We also convert all the audio to the format MP3 with sampling rate $44.1k$ Hz for evaluating the proposed method under other audio format.

3.2. Feature Selection

We mention the two features, DFT and MFCCs, in section 2.3. In order to determine which feature that do well in distinguishing duplicated segments and original ones, and the experimental audios are made by letting same person repeat same words at different time. Table 1 shows the statistical result that the PCCs between every two audios. It is clear that DFT is better than MFCCs in copy-move detection because MFCCs are commonly used as features in speech recognition systems. They can simulate auditory perception of human's ear well. But when doing speech recognition, the same content of audio clips that one person repeat at different time will definitely be recognized as the same words. That is the reason why MFCCs are not suitable for digital audio copy-move detection. As for DFT, the feature of those segment are different, because the frequency components of the words are different, even these words sound very similar. So using DFT as feature can distinguish the duplicated segment from normal segments well. In the following experiment, DFT is selected as extracted feature of digital audio.

3.3. Threshold Selection and Parameter Settings

Threshold selection is important for the proposed method. When using the value that too large, it may omit some duplicated segments. On the contrary, when using the value that too small, it may cause many false detections. So, the value of threshold must be appropriately set. For each two segments, we calculate the value of PCCs to select a proper threshold. The results of audio database are shown in Table 2. The results show that the feature we select can effectively distinguish the duplicated segments and the normal segments. In the end, we decide to use 0.95 as the threshold, it can detect most of duplicated segments and only treat few normal segments as duplicated ones.

The parameter ± and ² are also need to be determined. As we described in Section 2.4 and Section 2.5, Changing the values of ± or ² has an influence on both accuracy and performance in the proposed method. So we use the 1000 original audio database in the proposed method with different values of ± and ², and the performance of both accuracy and time consumption are assessed

Table 2. Detection result under different feature

PCCs	Over 0.98	Over 0.97	Over 0.96	Over 0.95	Over 0.90
Duplicated	95.5%	98.2%	99.4%	99.8%	100%
Normal	0%	0%	0%	0.01%	0.02%

Table 3. Detection result under different α and β

±	²			
	10	25	50	100
25	91.2%	92.5%	92.1%	91.6%
50	96.3%	96.2%	95.9%	96.1%
100	97.9%	98.2%	98.3%	98.1%
200	98.3%	98.3%	98.4%	98.5%

Table 4. Time consumption under different α and β

±	²			
	10	25	50	100
25	19086s	19193s	19095s	19580s
50	32317s	32266s	32277s	32814s
100	53445s	53565s	53394s	54383s
200	88278s	87156s	86894s	90316s

for seeking the best value. The results are shown in Table 3 and Table 4. We can see from the results that ± has a greater impact than ² on accuracy and time consumption. The higher the value of ± is, the higher the accuracy is and the longer the time consumption is. The reason is that more comparisons cost more time, and duplicated segments may be detected in the rest of comparisons. Calculating the sum of ² feature points is the different case. Addition operation is super fast in computer, hence adding addition operations in multiples cost far less time than adding more comparison. And main features are in low frequency part, so adding more high-frequency feature points has less impact on accuracy. For ensuring higher accuracy and less time consumption, we select the value 100 for ± and 25 for ² when evaluating the performance and comparing with other methods.

3.4. Evaluation of Performance

The evaluation of the performance use the common criteria: precision, recall and F_1 score.

The definition of the precision, recall and F_1 score are as follow:

Table 5. Detection results of the proposed method under WAV format

	Precision	Recall	F_1
Original	99.3%	96.9%	98.1%
Adding noise	98.8%	95.9%	97.3%
Filtering	99.1%	96.3%	97.7%

$$precision = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$$

$$recall = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$$

and F_1 score combine both precision and recall, and it represent the overall performance of the methods, it can be defined as:

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

Table 5 and Table 6 are the results of detection under the criteria. The results under WAV format represent that the proposed method is effective in copy-move forgery detection in audio. But, in the result under MP3 format, the precision is much lower than the result under WAV format. We consider the reason might be the MP3 format is a lossy format, some information is lost when the audios convert to MP3 format, therefore, more self-like segments are regarded as forgery. However, the proposed method still has a high result of recall.

Moreover, the time consumption is also another important criterion for evaluating the performance. Comparing with the other methods, the proposed method reduces the time complexity of detection as we mentioned. So, we also calculate the average time consumption of detection for evaluating this. Table 7 shows the results of time consumption and demonstrate the better efficiency of the proposed method.

3.5. Comparisons With Other Method

For evaluating the performance of the proposed method, we also compare the proposed method with other copy-move detection methods under WAV format. As we pointed out earlier, in the area of copy-move forgery detection of audio, few researches are made until now. So, we just compare our method with Yan's (Yan, Yang & Huang, 2015) method. The comparison uses the same evaluation criteria in Section 3.4 and also the same audio database. As the detection results shows in the Table 8, the proposed method has a better value of precision than Yan's method, and so is the recall and F_1 score. Besides, as shown in Table 9, time consumption of proposed method is also lower than Yan's method. And we found the shorter time length of each audio segment may cause the low performance of Yan's method.

In a word, the results of the comparison can demonstrate the good performance of the proposed method, it is more effective and robust than other state-of-the-art method.

Table 6. Detection results of the proposed method under MP3 format

	Precision	Recall	F_1
Original	74.38%	98.75%	84.85%
Adding noise	65.22%	98.00%	78.32%
Filtering	71.01%	99.25%	82.79%

Table 7. Average time consumption under different duration

Duration	1 min	10 min	30 min	60 min
Average time consumption	15.6s	153.4s	463.1s	998.2s

Table 8. Detection results under different method

		Precision	Recall	F_1
Original	Yan	45.9%	84.9%	59.6%
	Proposed	99.3%	96.9%	98.1%
Adding noise	Yan	44.5%	80.3%	57.3%
	Proposed	98.8%	95.9%	97.3%
Filtering	Yan	46.0%	85.2%	96.3%
	Proposed	99.1%	96.3%	97.7%

Table 9. Average time consumption under different method

	Yan	Proposed
1 min	18.8s	15.6s
10 min	197.7s	153.4s
30 min	632.9s	463.1s
60 min	1384.3s	998.2s

4. CONCLUSION

In this paper, we have proposed a method to detect copy-move forgery in digital audio fast and effectively. Firstly, the input audio is divided into syllables by auto segment, which make the proposed method more suitable for the human voice audio. Next, the DFT feature is extracted from the audio segment for similarity computation. Then we add an additional sorting step for reducing the redundant comparisons. Finally, we calculate the PCCs between one segment and some neighboring segments, the copy-move forgery can be detected by comparing the values with the threshold. The experiments show that the proposed method can detect copy-move forgery more accurately and effectively when comparing with another method, even the audio have been added noise or filtered. In the future, we will pay more attention to segmenting the audio more precisely and choosing more useful and robust features for improving the accuracy.

This work is supported by the National Natural Science Foundation of China (No. U1736118), the Natural Science Foundation of Guangdong (No. 2016A030313350), the Special Funds for Science and Technology Development of Guangdong (No. 2016KZ010103), the Key Project of Scientific Research Plan of Guangzhou (No. 201804020068), the Fundamental Research Funds for the Central Universities (No. 16lgjc83 and No. 17lgjc45).

REFERENCES

Bassia, P., Pitas, I., & Nikolaidis, N. (2001). Robust audio watermarking in the time domain. *Multimedia IEEE Transactions on*, *3*(2), 232–241. doi:10.1109/6046.923822

Chen, J., Xiang, S., Liu, W., & Huang, H. (2013). Exposing digital audio forgeries in time domain by using singularity analysis with wavelets. In *ACM Workshop on Information Hiding and Multimedia Security* (pp. 149-158). ACM. doi:10.1145/2482513.2482516

Cooper, A. J. (2010). Detecting butt-spliced edits in forensic digital audio recordings. In *39th International Conference: Audio Forensics: Practices and Challenges* (p. 1).

D'Alessandro, B., & Shi, Y. Q. (2009). Mp3 bit rate quality detection through frequency spectrum analysis. In *ACM Workshop on Multimedia and Security* (pp. 57-62). ACM. doi:10.1145/1597817.1597828

Farid, H. (1999). Detecting digital forgeries using bispectral analysis (MIT AI Memo AIM-1657). MIT.

Grigoras, C. (2007). Digital audio recording analysis: The electric network frequency (enf) criterion. *International Journal of Speech Language and the Law*, *12*(1), 63–76. doi:10.1558/sll.2005.12.1.63

Gupta, V., Boulianne, G., & Cardinal, P. (2010). Content-based audio copy detection using nearest-neighbor mapping. In *IEEE International Conference on Acoustics Speech and Signal Processing* (pp. 261-264). IEEE. doi:10.1109/ICASSP.2010.5495963

Huijbregtse, M., & Geradts, Z. (2009). Using the ENF Criterion for Determining the Time of Recording of Short Digital Audio Recordings. In *International Workshop on Computational Forensics* (pp. 116-124). Springer-Verlag. doi:10.1007/978-3-642-03521-0_11

Kraetzer, C., Oermann, A., Dittmann, J., & Lang, A. (2007). Digital audio forensics:a first practical evaluation on microphone and environment classification. In *The Workshop on Multimedia & Security* (pp.63-74). DBLP.

Li, W., Xue, X., & Lu, P. (2006). Localized audio watermarking technique robust against time-scale modification. *IEEE Transactions on Multimedia*, *8*(1), 60–69. doi:10.1109/TMM.2005.861291

Ouali, C., Dumouchel, P., & Gupta, V. (2015). Efficient spectrogram-based binary image feature for audio copy detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing* (pp.1792-1796). IEEE. doi:10.1109/ICASSP.2015.7178279

Pan, X., Zhang, X., & Lyu, S. (2012). Detecting splicing in digital audios using local noise level estimation. In *IEEE International Conference on Acoustics, Speech and Signal Processing* (Vol.22, pp.1841-1844). IEEE. doi:10.1109/ICASSP.2012.6288260

Pfitzinger, H. R., Burger, S., & Heid, S. (1996). Syllable detection in read and spontaneous speech. In *Proceedings International Conference on Spoken Language ICSLP '96* (Vol. 2, pp.1261-1264). IEEE. doi:10.1109/ICSLP.1996.607838

Shen, J. L., Hung, J. W., & Lee, L. S. (1998). Robust entropy-based endpoint detection for speech recognition in noisy environments. In *International Conference on Spoken Language Processing, Incorporating the, Australian International Speech Science and Technology Conference*, Sydney Convention Centre, Sydney, Australia.

Wang, X. Y., & Zhao, H. (2006). A novel synchronization invariant audio watermarking scheme based on dwt and dct. *IEEE Transactions on Signal Processing*, *54*(12), 4835–4840. doi:10.1109/TSP.2006.881258

Wu, C. P., Su, P. C., & Kuo, C. C. J. (2000). Robust and efficient digital audio watermarking using audio content analysis. In *Proceedings of SPIE - The International Society for Optical Engineering*. doi:10.1117/12.384992

Xiang, S., & Huang, J. (2007). Histogram-based audio watermarking against time-scale modification and cropping attacks. *IEEE Transactions on Multimedia*, *9*(7), 1357–1372. doi:10.1109/TMM.2007.906580

Xiao, J. N., Jia, Y. Z., Fu, E. D., Huang, Z., Li, Y., & Shi, S. P. (2014). Audio authenticity: Duplicated audio segment detection in waveform audio file. *Journal of Shanghai Jiaotong University(Science)*, *19*(4), 392–397. doi:10.1007/s12204-014-1515-5

Yan, Q., Yang, R., & Huang, J. (2015). Copy-move detection of audio recording with pitch similarity. In *IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 1782-1786). IEEE. doi:10.1109/ICASSP.2015.7178277

Yang, R., Qu, Z., & Huang, J. (2008). Detecting digital audio forgeries by checking frame offsets. In *The Workshop on Multimedia & Security* (pp. 21-26). ACM. doi:10.1145/1411328.1411334

Yang, R., Qu, Z., & Huang, J. (2012). Exposing mp3 audio forgeries using frame offsets. *ACM Transactions on Multimedia Computing Communications and Applications*, *8*(2S), 1–20.

Xinchao Huang received the B.S. degree in Software Engineering from Northeastern University, Shenyang, China in 2015. He is currently working toward the M.S. degree in Software Engineering at School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China. His research interests include multimedia forensics and security.

Zihan Liu graduated from Sun Yat-sen University in June 2017 and received Bachelor of Engineering degree. Now, she is pursuing Master of Science degree in Northeastern University. Her research interest is digital forensics.

Wei Lu received a B.S. degree in Automation from Northeast University, China in 2002, a M.S. degree and a Ph.D. degree in Computer Science from Shanghai Jiao Tong University, China in 2005 and 2007, respectively. He was a research assistant at Hong Kong Polytechnic University from 2006 to 2007. He is currently an Associate Professor in the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China. His research interests include multimedia forensics and security, multimedia signal processing, image/video intelligent analysis. Wei Lu is the corresponding author.