Hindawi
Security and Communication Networks
Volume 2021, Article ID 6659371, 6 pages
https://doi.org/10.1155/2021/6659371

Research Article

Exposing Speech Transsplicing Forgery with Noise Level Inconsistency

Diqun Yan, Mingyu Dong, and Jinxing Gao

Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315211, China

Correspondence should be addressed to Diqun Yan; yandiqun@nbu.edu.cn

Received 7 December 2020; Revised 8 January 2021; Accepted 13 January 2021; Published 27 January 2021

Academic Editor: Manjit Kaur

Splicing is one of the most common tampering techniques for speech forgery in many forensic scenarios. Some successful approaches have been presented for detecting speech splicing when the splicing segments have different signal-to-noise ratios (SNRs). However, when the SNRs between the spliced segments are close or even same, no effective detection methods have been reported yet. In this study, noise inconsistency between the original speech and the inserted segment from other speech is utilized to detect the splicing trace. First, noise signal of the suspected speech is extracted by a parameter-optimized noise estimation algorithm. Second, the statistical Mel frequency features are extracted from the estimated noise signal. Finally, the spliced region is located by utilizing a change point detection algorithm on the estimated noise signal. The effectiveness of the proposed method is evaluated on a well-designed speech splicing dataset. The comparative experimental results show that the proposed algorithm can achieve better detection performance than other algorithms.

1. Introduction

With the wide spread of social networks and the rapid development of powerful audio editing tools (such as Adobe Audition and GoldWave), digital speech can be easily accessed, manipulated, and distributed. Such tools have provided lots of convenience in various aspects such as social activity, news media, entertainment, and so forth. These modified speeches, however, may cause unpredictable results when they are presented in a scene such as justice or criminal investigation. Digital speech forensics [1–3] is a valuable technique for determining the authenticity of digital speech. By analyzing the modification traces left in the suspected speech, digital forensics can identify the tampering type and locate the tampering position [4].

Deletion, insertion, and splicing are three most commonly tampering operations that can significantly change the content of the original speech. Splicing is an operation in which one or more speech segments are inserted in the original one to change the content of the target speech. In general, splicing is always accompanied by deletion and insertion. According to whether the inserted speech segment is from the original speech or not, splicing can be further divided into self-splicing and transsplicing, respectively. Specifically, self-splicing refers to copying a segment in the original speech and inserting it into the other region in the same speech. Since the self-splicing will introduce high-similarity regions in the spliced speech, the detector can take the similarity of speech features as criterion to find the splicing matching regions. In real scenarios, transsplicing is relatively more common than self-splicing. On the one hand, the forgers tend to splice speech components from different source/scenes. On the other hand, it is a hard task for the forgers to find the splicing segment from the original speech in most cases. In this work, we focus on the detection of speech transsplicing.

As an important branch of multimedia security [5, 6], many splicing detection algorithms [7–9] for digital speech have been proposed over the last decade. The ENF- (electric network frequency) based method [10] is effective for detecting speech splicing, in which the ENF signal is extracted from a questioned audio recording and matches it with the reference signal in an ENF database. Reis et al. [11] proposed an ESPRIT-Hilbert ENF estimator with an outlier

detector based on the kurtosis of the estimated ENF. Then, the kurtosis is taken as an input for a support vector machine classifier to indicate the presence of splicing. However, ENF-based detection algorithms may not be applicable when the speech is recorded with the well-designed or battery-operated devices. On the other hand, the reference ENF dataset is needed during an ENF-based forensic investigation process. Imran [12] proposed a splicing detection algorithm based on intrinsic statistical properties of suspected speech. The speech is first divided into segments using voice activity detection, and the histogram of one-dimensional LBP (local binary pattern) is exploited as the detection feature. Zhao et al. [13] introduced channel impulse response to detect speech splicing. The impulse response amplitude and background noise are used to determine the location of the splicing.

In real scenarios, in order to remove the splicing trace, the forger would try best to keep the SNR (signal-noise ratio) of the processed speech as consistent as possible between the spliced and the original regions. This will greatly increase the difficulty of the splicing detection task. As far as we know, there is no prior work on transsplicing detection with the same SNR. In this study, we proposed an approach for detecting transsplicing with the same SNR. First, the Sorensen algorithm [14] is utilized to estimate the noise level of the suspected speech. Then, the variances of Mel frequency cepstral coefficient (MFCC) [15] for estimated noise signal are calculated as the detecting features. Finally, the spliced region is located by a change point detection algorithm based on the penalty cost function [16]. The performance of the proposed algorithm is evaluated on a well-designed speech splicing dataset. The experimental results show that the proposed algorithm achieves better detection accuracy compared with other algorithms.

The rest of the study is organized as follows. The main work of this study is described in Section 2, in which noise estimation, feature extraction, and the change point detection algorithm are described in detail. Section 3 will present the splicing dataset and the experimental results. Finally, the conclusion is drawn in Section 4.

2. Proposed Transsplicing Detection Algorithm

The proposed framework for transsplicing detection and localization is shown in Figure 1. First, the Sorensen algorithm is adopted to estimate the noise signal. Next, the estimated noise is framed, and its Mel-frequency cepstral coefficients are extracted. The variance of the coefficients is calculated as the detecting feature. Finally, the change point detection algorithm is applied on the variance sequence to detect and locate the splicing.

2.1. Noise Estimation. Sorensen [14] proposed a recursive averaging noise estimation algorithm. The idea is that different attenuation rules are adopted to different regions to estimate the noise in the speech accurately. Figure 2 shows the flowchart of this algorithm.

Let $y(i)$ be the suspected speech at time i, which consists of clean speech $s(i)$ and additive noise $n(i)$. First, the windowed and framed speech signal is subjected to short-time Fourier transform (STFT):

$$Y(\lambda, k) = S(\lambda, k) + N(\lambda, k), \tag{1}$$

where $\lambda \in Z$ is the time index, $k \in \{0, 1, \ldots, K-1\}$ is the frequency bin index, L is the window length, and $S(\lambda, k)$ and $N(\lambda, k)$ are the STFT coefficients of $s(i)$ and $n(i)$, respectively.

Then, the periodograms P_Y can be calculated as

$$P_Y(\lambda, k) \triangleq |Y(\lambda, k)|^2. \tag{2}$$

Next, P_Y is spectrally smoothed to produce $p_Y(\lambda, k)$ and then temporally smoothed to $p(\lambda, k)$. Then, the temporal minimum values $p_{\min}(\lambda, k)$ could be tracked within a minimum search window of length $D_{\min}$, that is,

$$p_{\min}(\lambda, k) = \min\left(p(\psi, k) \mid \lambda - D_{\min} < \psi \le \lambda\right), \tag{3}$$

where $\psi \in Z$, and $D_{\min} = U * V$. Window D represents an analysis window length. Since it is computationally expensive to find minimum in each frequency band for each frame, an efficient procedure [17] is proposed in which the analysis window is divided into U subwindows of V samples. Hence, the minimum is updated for every V samples, stored it for later use, and reduced the number of comparation operations per frame and frequency bin on $1 + (U-1)V$.

For $D(\lambda, k) = 1$, the noise periodogram estimation is equal to a time-varying power scaling of the minimum tracks $p_{\min}(\lambda, k)$. For $D(\lambda, k) = 0$, it is equal to the noisy speech periodogram $P_Y(\lambda, k)$, that is,

$$P_{\widehat{N}}(\lambda, k) = \left\{ \begin{array}{ll} R_{\min}(\lambda) p_{\min}(\lambda, k), & \text{if } D(\lambda, k) = 1, \\ P_Y(\lambda, k), & \text{if } D(\lambda, k) = 0, \end{array} \right\}, \tag{4}$$

where $D(\lambda, k)$ is used to determine whether speech exists. $R_{\min}(\lambda)$ is a bias compensation factor, and it only updates in the nonspeech frames.

A smooth estimate of the noise magnitude spectrum can be obtained by

$$|\widehat{N}(\lambda, k)| = \sqrt{\widetilde{P}_{\widehat{N}}(\lambda, k)}. \tag{5}$$

After the above steps, we obtained the enhanced speech $\widehat{s}(i)$. Finally, the estimated noise signal $\widehat{n}(i)$ can be obtained by subtracting the enhanced speech $\widehat{s}(i)$ from the noisy speech $y(i)$, that is,

$$\widehat{n}(i) = y(i) - \widehat{s}(i). \tag{6}$$

It is seen from equation (3) that $D_{\min}$ plays an important role in the noise estimation process. $D_{\min}$ is mainly used to control a fixed-length window. In the noise estimation process of each frame, the minimum $p_{\min}(\lambda, k)$ in the window is tracked, and the value obtained by the tracking is used to continuously update $p_{\min}(\lambda, k)$. Finally, the noise power spectrum $P_{\widehat{N}}(\lambda, k)$ is calculated by $p_{\min}(\lambda, k)$. It can be seen from the above analysis that reasonable adjustment

Figure 1: Framework of proposed splicing detection and localization.

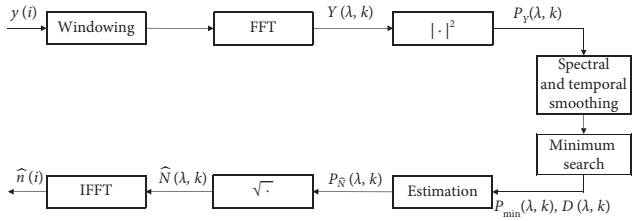

Figure 2: Flowchart of the Sorensen noise estimation algorithm.

of U and V can effectively improve the noise estimation performance of the algorithm.

2.2. Detection Feature Extraction. For each frame of the estimated noise, Mel frequency cepstral coefficients are extracted, which is based on the human peripheral auditory system. Figure 3 shows the diagram of MFCC extraction.

First, the estimated noise signal $\widehat{n}(i)$ is subjected to DFT to obtain a linear spectrum $\widehat{N}(k)$. Then, $\widehat{N}(k)$ is filtered by the Mel frequency filter bank $H_m(k)$ to obtain the Mel spectrum. In order to make the result more robust to noise and spectral estimation errors, the logarithmic energy of the Mel spectrum is generally taken, that is,

$$L(m) = \ln\left[\sum_{k=1}^{N} |\widehat{N}(k)|^2 H_m(k)\right], \quad m = 1, 2, \ldots, M, \tag{7}$$

where m is the number of filter banks.

Next, $L(m)$ is subjected to DCT to obtain the MFCC coefficient:

$$mfcc(j) = \sum_{m=1}^{M} L(m)\cos\left(n(m-0.5)\frac{\pi}{m}\right), \quad (1 \le j \le J), \tag{8}$$

where j is the index of the cepstral coefficients.

Finally, for each frame, the variance of $mfcc(j)$ can be calculated by equation (9), and we can obtain a variance sequence for each suspected speech.

$$V = \frac{1}{J}\sum_{j=1}^{J} (mfcc(j) - \overline{mfcc})^2. \tag{9}$$

2.3. Change Point Detection. Since the segments of trans-splicing come from the different sources/scenes, we consider the inconsistencies of the noise characteristics mixed in the suspected speech to be a clue of splicing. It means that there will be a change on noise characteristics where the splicing happened. Hence, the splicing detection and localization can be transformed into a change point detection problem. Algorithms for change points' detection [18–20] have made good progress in recent years. Lavielle [16] proposed a model selection method based on a penalized contrast which is applied to the change point problem. It can be used for estimating the number of change points and their location. In this work, Lavielle's algorithm is adopted to find the splicing positions.

Let $V = (V_1, V_2, \ldots, V_n)$ be the variance sequence of estimated noise signal and K be some integer. Similarly, let $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_{K-1})$ be a sequence of integers satisfying $0 < \alpha_1 < \alpha_2 < , \cdots, < \alpha_{K-1} < n$. For any $1 \le k \le K$, let $M(V_{\alpha_{k-1}+1}, V_{\alpha_{k-1}+2}, \ldots, V_{\alpha_K}; \beta)$ be a contrast function for estimating the unknown true value of the parameter β in the segment k. It means that there will be an estimated value of β ($\widehat{\beta}$) when the contrast function reaches it minimum. In other words, the minimum contrast estimate $\widehat{\beta}(V_{\alpha_{k-1}+1}, V_{\alpha_{k-1}+2}, \ldots, V_{\alpha_K})$, computed on segment k of α, is defined as a solution of the following minimization problem:

$$M\left(V_{\alpha_{k-1}+1}, V_{\alpha_{k-1}+2}, \ldots, V_{\alpha_K}; \widehat{\beta}\left(V_{\alpha_{k-1}+1}, V_{\alpha_{k-1}+2}, \ldots, V_{\alpha_K}\right)\right) \le M\left(V_{\alpha_{k-1}+1}, V_{\alpha_{k-1}+2}, \ldots, V_{\alpha_K}; \beta\right). \tag{10}$$

Then, we define the contrast function $J(\alpha, v)$ as

$$J(\alpha, s) = \frac{1}{n}\sum_{k=1}^{K} M\left(V_{\alpha_{k-1}+1}, V_{\alpha_{k-1}+2}, \ldots, V_{\alpha_K}; \widehat{\beta}\left(V_{\alpha_{k-1}+1}, V_{\alpha_{k-1}+2}, \ldots, V_{\alpha_K}\right)\right), \tag{11}$$

where $\alpha_0 = 0$, $\alpha_K = n$.

As an example, consider the flowing model:

$$V_i = \mu_i + \sigma_i \varepsilon_i, \quad (1 \le i \le n), \tag{12}$$

where ε_i is a sequence with zero mean and unit variance. In the case of changes in the variance, μ_i is a constant sequence and σ_i is a piecewise one. The contrast function can be defined as a Gaussian log-likelihood, even if ε_i is not a Gaussian sequence.

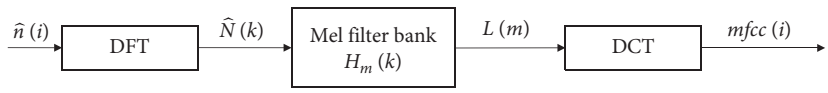

Figure 3: Diagram of MFCC extraction.

$$M\left(V_{\alpha_{k-1}+1}, V_{\alpha_{k-1}+2}, \ldots, V_{\alpha_K}; \sigma^2\right) = (\alpha_k - \alpha_{k-1})\log\left(\sigma^2\right) + \frac{1}{\sigma^2}\sum_{i=\alpha_{k-1}+1}^{\alpha_K}(V_i - \mu)^2. \tag{13}$$

Then,

$$\mathrm{J}(\alpha, v) = \frac{1}{n}\sum_{k=1}^{K}(\alpha_k - \alpha_{k-1})\log\left(\widehat{\tau}^2_{\alpha_{k-1}+1:\ \alpha_k}\right), \tag{14}$$

where $\widehat{\tau}^2_{\alpha_{k-1}+1:\ \alpha_k} = (1/(\alpha_k - \alpha_{k-1}))\sum_{i=\alpha_{k-1}+1}^{\alpha_K}(V_i - \overline{V})^2$ is the variance of $(V_{\alpha_{k-1}+1}, V_{\alpha_{k-1}+2}, \ldots, V_{\alpha_k})$. For instance, when the maximum number of segments $K_{\max} = 3$, the number of change points is $K_{\max} - 1 = 2$, and the change boundary is (α_1, α_2).

Finally, we summarize our splicing detection algorithm as follows. First, we estimate the power spectral density $P_{\widehat{N}}(\lambda, k)$ of the noise in the noisy speech signal $y(i)$ and then use $P_{\widehat{N}}(\lambda, k)$ to obtain the enhanced speech signal $\widehat{s}(i)$. Therefore, the noise signal $\widehat{n}(i)$ can be estimated with the noisy speech $y(i)$ and the enhanced speech $\widehat{s}(i)$. Then, the estimated noise $\widehat{n}(i)$ is framed and windowed, and then for each frame, M-dimensional MFCC coefficients are calculated. The variance sequence $V = (V_1, V_2, \ldots, V_n)$ of MFCC coefficients is obtained and taken as the input of the change point detection algorithm, and then, the penalty cost function is constructed by equation (11). Finally, the estimated parameters of the penalty cost function $K^* - 1$ and $(\alpha_{K^*-2}, \alpha_{K^*-1})$ represent the number of change points and the boundaries of the change segments, respectively. Among them, the boundary of the change segment is the final detected tampering position.

3. Experimental Results

In this section, we first describe the dataset adopted in this work. Additionally, as mentioned in subsection 2.1, the performance of the proposed detection algorithm depends strongly on the effectiveness of the noise estimation. Hence, the noise estimation algorithm is evaluated to find the optimal parameters. Then, the performance of the proposed splicing detection method with optimal noise estimation is present.

3.1. Splicing Dataset. The transsplicing speech samples in this study are created based on NOIZEUS speech corpus [21] which is derived from the clean speech contaminated by various kinds of noise in the real world. The clean speech comes from 30 IEEE statements containing three male and three female pronunciations. The noise signals in NOIZEUS come from the AURORA-2 database [22], including noise from train stations, airports, exhibition halls, streets, and restaurants, as well as car noise, noise from commuter trains, and babble noise from multiperson speech. During noise contamination, various SNR cases including 0 dB, 5 dB, 10 dB, and 15 dB have been considered.

The creation process of the splicing speech dataset is as follows. First, the samples of NOIZEUS corpus are divided into two classes: the original samples and the samples to be spliced. Then, for each sample to be spliced, we further cut it into 4 different segments by using random numbers. For each original sample, a pseudorandom generator is used to determine where the segment will be spliced. Next, the splicing is performed, and the spliced speech is saved with the original sampling rate. In this work, the SNR of the original sample is kept the same as the segment to be spliced.

In the experiment, there will be 42 types of samples in each splicing subset, and each type contains 30 samples. As a result, there will be 1260 samples in each splicing subset. Each sample is 8 KHz, mono, 16 bit quantized, and the duration is 3-4 seconds.

3.2. Performance Evaluation on Noise Estimation. It can be seen from the analysis in Section 2.1 that the parameters U and V will affect the performance of the noise estimation algorithm. In order to find the optimal U and V, we first adjust the U and V values in the Sorensen algorithm to estimate the noise of 1260 segments of each subset and then calculate the average SNR of the 1260-segment speech under each U and V case. The experimental results for 0 dB and 5 dB speech are given in Tables 1 and 2.

It can be clearly seen from Tables 1 and 2 that U and V have a great influence on the performance of the Sorensen algorithm. For example, the estimation error for 0 dB case is minimized at −0.0737 dB when (U, V) is (2, 5). And the best choice for 5 dB case is (4, 4). Table 3 gives the optimal U and V for various SNR cases.

Additionally, we compared the optimized Sorensen algorithm with other typical noise estimation algorithms. From Table 4, the optimized algorithm achieves the best estimated results in various SNR cases.

3.3. Performance Evaluation on Splicing Detection. In MFCC extraction, we set the number of filters m to 27 and the number of cepstral coefficients J to 12. For Lavielle's algorithm [16], we set the maximum number of segments $K_{\max}$ to 3 and only variance change is considered.

Table 1: Noise estimation for 0 dB.

		V						
		8	7	6	5	4	3	2
U	5	1.9714	1.7684	1.5395	1.2562	0.8877	0.4256	−0.2804
	4	1.6638	1.4731	1.2296	0.9308	0.5658	0.0819	−0.6422
	3	1.2830	1.0713	0.8185	0.5161	0.1296	−0.3664	−1.0596
	2	0.7330	0.5187	0.2536	**−0.0737**	−0.4621	−0.9503	−1.5494

The value in bold is used to emphasise that the noise estimation algorithm achieves the best performance when (U, V) is (2, 5).

Table 2: Noise estimation for 5 dB.

		V						
		8	7	6	5	4	3	2
U	5	7.0750	6.8253	6.5178	6.1066	5.5315	4.7870	3.5437
	4	6.6826	6.4245	6.0547	5.5989	**5.0011**	4.1934	2.8933
	3	6.1350	5.8040	5.4229	4.9272	4.2658	3.4006	2.1375
	2	5.2449	4.9374	4.4928	3.8981	3.1890	2.3344	1.2947

The value in bold type is used to emphasise that the noise estimation algorithm achieves the best performance when (U, V) is (4, 4).

Table 3: Optimal parameters in various SNRs.

	SNR (dB)			
	0	5	10	15
U	2	4	3	4
V	5	4	7	7

F score is introduced as an objective metric to evaluate the performance of the proposed algorithm, which can be expressed as follows:

$$F = \frac{(2 * \text{precision} * \text{recall})}{(\text{precision} + \text{recall})},$$

$$\text{Precision} = \frac{\widetilde{\chi} \cap \chi}{\widetilde{\chi}}, \tag{15}$$

$$\text{Recall} = \frac{\widetilde{\chi} \cap \chi}{\chi},$$

where precision is the accuracy rate, recall is the recall rate, χ is the actual splicing region, and $\widetilde{\chi}$ is the detected splicing region. It can be seen from equation (15) that the larger the F value, the better the detection capability of the algorithm.

As a comparison to [7, 9], we adopt the optimal parameters in Table 3 to detect the splicing trace. The F scores are shown in Table 5. It can be seen that the proposed method achieves better detection performance in all SNR cases. Meanwhile, it can be seen from Table 3 that the detection performance of the algorithm gradually deteriorates with the SNR increases. This is consistent with the situation in the actual scene, that is, the lower the noise energy contained in the speech signal, the more difficult the noise estimation algorithm is to extract the noise. In addition, according to the results in Tables 3 and 5, the detection result of the algorithm tends to become better with the decrease of U and V. It indicates that the speed of the noise estimation will be beneficial to improve the detection rate of the algorithm.

Table 4: SNR estimation for various algorithms.

Algorithm	0 dB	5 dB	10 dB	15 dB
[23]	4.7419	9.8905	14.7146	18.5088
[24]	2.6064	7.7292	12.5088	16.5755
[25]	3.5518	8.2249	12.3533	15.6897
[26]	3.3987	8.7689	13.7923	17.8140
[27]	4.3277	8.5393	12.1390	14.7518
Optimized Sorensen	**0.1296**	**5.0011**	**10.0396**	**15.0139**

The value in bold type is used for emphasis that the performance of the optimized algorithm is better than other algorithms.

Table 5: F scores of splicing detections.

Algorithms	SNR cases (dB)			
	0-0 dB	5-5 dB	10-10 dB	15-15 dB
[7]	0.7459	0.7317	0.6734	0.6605
[9]	0.7924	0.7999	0.7805	0.7672
Proposed	**0.8302**	**0.8137**	**0.7923**	**0.7685**

The values in bold type is used for emphasis that the performance of the proposed algorithm is better than other two algorithms.

4. Conclusion and Future Work

In this study, a novel method for the speech transsplicing detection algorithm has been proposed. Considering that the segment to be spliced and the original segment have different noise levels, the noise of the suspected speech is estimated first. Then, we extract the variance of the 12-dimensional MFCC coefficients from the estimated noise and utilize the change point detection algorithm based on the penalty cost function to locate the splicing region, finding that the variance of the spliced region is significantly lower than that of the nonspliced regions. Experimental results show that the detection algorithm can accurately determine the starting position of splicing and can detect the entire splicing region. Compared with the splicing detection methods based on grid frequency and intrinsic statistical law of speech, the proposed method has fewer assumptions and can be applied to more forensic scenarios. The future work will focus on extracting more efficient hybrid features to further improve detection accuracy, and more scenarios closer to the real world such as reverberation will be considered.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 61300055), Ningbo Natural Science Foundation (Grant No. 202003N4089), and K. C. Wong Magna Fund in Ningbo University.

References

[1] Q. Yan, R. Yang, and J. Huang, "Detection of speech smoothing on very short clips," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 9, pp. 2441–2453, 2019.

[2] D. Luo, R. Yang, B. Li, and J. Huang, "Detection of double compressed AMR audio using stacked autoencoder," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 2, pp. 432–444, 2017.

[3] T. Bianchi, A. Rosa, M. Fontani, G. Rocciolo, and A. Piva, "Detection and localization of double compression in MP3 audio tracks," *EURASIP Journal on Information Security*, vol. 2014, pp. 1–14, 2014.

[4] L. Verdoliva, "Media forensics and DeepFakes: an overview," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 5, pp. 910–932, 2020.

[5] M. Kaur, D. Singh, K. Sun, and U. Rawat, "Color image encryption using non-dominated sorting genetic algorithm with local chaotic search based 5D chaotic map," *Future Generation Computer Systems*, vol. 107, pp. 333–350, 2020.

[6] A. Gupta, D. Singh, and M. Kaur, "An efficient image encryption using non-dominated sorting genetic algorithm-III based 4-D chaotic maps," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 3, pp. 1309–1324, 2020.

[7] X. Meng, C. Li, and L. Tian, "Detecting audio splicing forgery algorithm based on local noise level estimation," in *Proceedings of the 2018 5th International Conference on Systems and Informatics (ICSAI)*, pp. 861–865, Nanjing, China, November 2018.

[8] X. Lin and X. Kang, "Supervised audio tampering detection using an autoregressive model," in *Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2142–2146, New Orleans, LA, USA, March 2017.

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

[10] A. Hajj-Ahmad, R. Garg, and M. Min Wu, "Spectrum combining for ENF signal estimation," *IEEE Signal Processing Letters*, vol. 20, no. 9, pp. 885–888, 2013.

[11] P. M. G. I. Reis, J. P. C. Lustosa da Costa, R. K. Miranda, and G. Del Galdo, "ESPRIT-Hilbert-based audio tampering detection with SVM classifier for forensic analysis via electrical network frequency," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 853–864, 2017.

[12] M. Imran, Z. Ali, S. Bakhsh, and S. Akram, "Blind detection of copy-move forgery in digital audio forensics," *IEEE Access*, vol. 5, pp. 12843–12855, 2007.

[13] H. Zhao, Y. Chen, R. Wang, and H. Malik, "Audio splicing detection and localization using environmental signature," *Multimedia Tools and Applications*, vol. 76, no. 12, pp. 13897–13927, 2017.

[14] K. Sorensen and S. Andersen, "Speech enhancement with natural sounding residual noise based on connected time-frequency speech presence regions," *EURASIP Journal on Advances in Signal Processing*, vol. 2005, pp. 2954–2964, 2005.

[15] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[16] M. Lavielle, "Using penalized contrasts for the change-point problem," *Signal Processing*, vol. 85, no. 8, pp. 1501–1510, 2005.

[17] R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statistics," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 5, pp. 504–512, 2001.

[18] F. Desobry, M. Davy, and C. Doncarli, "An online kernel change detection algorithm," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2961–2974, 2005.

[19] L. I. Kuncheva and W. J. Faithfull, "PCA feature extraction for change detection in multidimensional unlabeled data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 1, pp. 69–80, 2014.

[20] S. Liu, M. Yamada, N. Collier, and M. Sugiyama, "Change-point detection in time-series data by relative density-ratio estimation," *Neural Networks*, vol. 43, pp. 72–83, 2013.

[21] Y. Hu and P. Loizou, "Subjective comparison of speech enhancement algorithms," *Speech Communication*, vol. 49, no. 7-8, pp. 588–601, 2006.

[22] H. Hirsch and D. Pearce, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noise conditions," in *Proceedings of the Sixth International Conference on Spoken Language Processing, ICSLP 2000/INTERSPEECH 2000*, pp. 29–32, Beijing, China, October 2000.

[23] H. G. Hirsch and C. Ehrlicher, "Noise estimation techniques for robust speech recognition," in *Proceedings of the 1995 International Conference on Acoustics, Speech and Signal Processing*, pp. 153–156, Detroit, MI, USA, May 1995.

[24] I. Cohen and B. Berdugo, "Noise estimation by minima controlled recursive averaging for robust speech enhancement," *IEEE Signal Processing Letters*, vol. 9, no. 1, pp. 12–15, 2002.

[25] S. Rangachari and P. C. Loizou, "A noise-estimation algorithm for highly non-stationary environments," *Speech Communication*, vol. 48, no. 2, pp. 220–231, 2006.

[26] I. Cohen, "Noise spectrum estimation in adverse environments: improved minima controlled recursive averaging," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 5, pp. 466–475, 2003.

[27] G. Doblinger, "Computationally efficient speech enhancement by spectral minima tracking in subbands," *Proceedings of Eurospeech*, vol. 2, pp. 1513–1516, 1995.