Hindawi
Scientific Programming
Volume 2021, Article ID 5595850, 10 pages
https://doi.org/10.1155/2021/5595850

Research Article

Frame Duplication Forgery Detection and Localization Algorithm Based on the Improved Levenshtein Distance

Honge Ren,[1,2] **Walid Atwa**,[3] **Haosu Zhang**,[2] **Shafiq Muhammad**,[4] **and Mahmoud Emam**[5]

[1]*College of Information and Computer Engineering, Northeast Forestry University, Harbin 150040, China*
[2]*Heilongjiang Forestry Intelligent Equipment Engineering Research Center, Harbin 150040, China*
[3]*Department of Computer Science, Faculty of Computers and Information, Menoufia University, Shebin El-Koom 32511, Egypt*
[4]*Cyberspace Institute of Advance Technology, Guangzhou University, Guangzhou, China*
[5]*Department of Mathematics and Computer Science, Faculty of Science, Menoufia University, Shebin El-Koom 32511, Egypt*

Correspondence should be addressed to Honge Ren; nefu_rhe@163.com and Mahmoud Emam; ma7moud_emam@yahoo.com

Received 12 January 2021; Revised 3 March 2021; Accepted 20 March 2021; Published 1 April 2021

Academic Editor: Sikandar Ali

In this digital era of technology and software development tools, low-cost digital cameras and powerful video editing software (such as Adobe Premiere, Microsoft Movie Maker, and Magix Vegas) have become available for any common user. Through these softwares, editing the contents of digital videos became very easy. Frame duplication is a common video forgery attack which can be done by copying and pasting a sequence of frames within the same video in order to hide or replicate some events from the video. Many algorithms have been proposed in the literature to detect such forgeries from the video sequences through analyzing the spatial and temporal correlations. However, most of them are suffering from low efficiency and accuracy rates and high computational complexity. In this paper, we are proposing an efficient and robust frame duplication detection algorithm to detect duplicated frames from the video sequence based on the improved Levenshtein distance. Extensive experiments were performed on some selected video sequences captured by stationary and moving cameras. In the experimental results, the proposed algorithm showed efficacy compared with the state-of-the-art techniques.

1. Introduction

In our daily life, digital videos are playing a vital role in many fields of applications such as surveillance systems, medical fields, and criminal investigations. Because of the availability of low-cost digital video cameras and powerful video editing tools (such as Adobe Premiere, Microsoft Movie Maker, and Magix Vegas), it is now easy for common users to edit the video contents without leaving any visual traces of forgeries. So, we cannot trust in the authenticity of such videos anymore. Therefore, the authentication of such videos is becoming a very important research area these days. Digital video forensics is an emerging research area which aims at validating the authenticity of such videos [1]. The classification of digital video forensics is shown in Figure 1, where it can be divided into 3 categories: identification of the source camera, discrimination of computer-generated videos, and video forgery detection (video tampering detection) [1].

Video forgery manipulations can be acted in the three domains: spatial domain (intraframe forgery), temporal domain (interframe forgery), and spatio-temporal domain. Intraframe forgeries may include region duplication (copy-move) and splicing inside the frame itself whereas interframe forgeries include frame duplication, frame insertion, frame shuffling, and frame deletion [1].

Digital video forgery detection algorithms aim to detect the traces of forgeries in the digital video sequence. As in digital images, digital video forgery detection techniques also can be classified into active and passive (blind) techniques. In the passive video forgery detection techniques, the authenticity of a forged video is verified without the existence

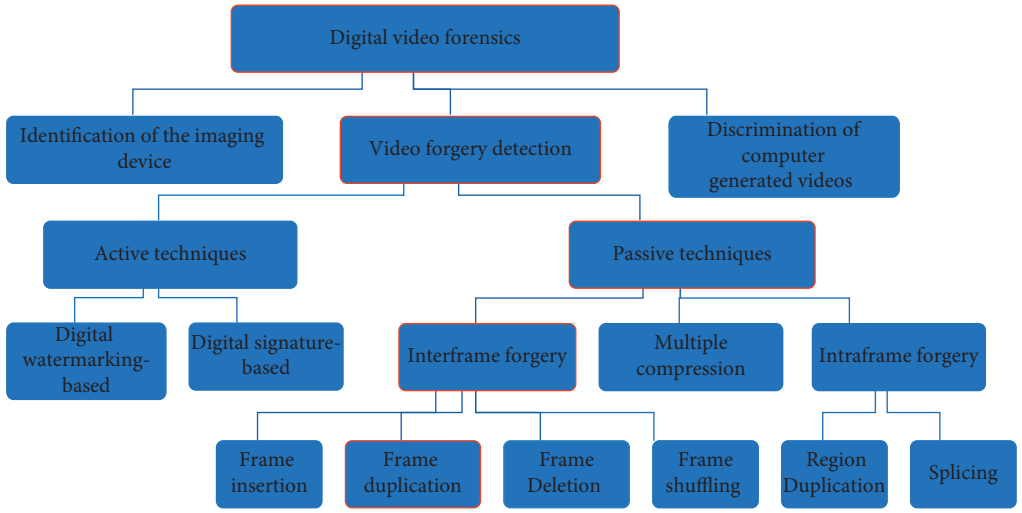


Figure 1: Digital video forensics and forgery detection techniques classification.

of the original video and only depends on extracting some features or footprints from the forged video which have been left by the editing operations [1]. These footprints may include the high spatio-temporal correlation among frame intensity values [2], noise and motion residues [3], artifacts in optical flow [4], motion-compensated edge artifact (MCEA) [5], and frames quality assessments [6] whereas active techniques require embedding information into the video such as digital watermarking [7], but this kind of techniques is not preferable by many researchers because it requires the existence of the original video along with the tampered one which is usually unavailable.

Frame duplication forgery is a common forgery types in digital videos, and it is an interframe forgery which occurs in the temporal domain. It can be performed by copying and pasting some frames in another location in the same video sequence in order to hide or replicate some events from the video. Figure 2 shows the process of frame duplication attack, where frames from 1 to 6 are copied and then pasted at another location instead of frames from 7 to 12 in order to remove the existence of a moving car crosses a parking area and passes behind a lamppost, without leaving any visual traces of forgeries. The original video example is taken from the LASIESTA dataset [8].

Cloning frames from the same video sequence raise the difficulty of frame duplication forgery detection, making it uneasy to detect color changes and illumination condition [9]. Although a variety of methods have been proposed, these methods still face the following challenges in frame duplication forgery:

(1) High computational complexity

(2) Low detection rate in the static scenes

(3) Unable to locate the location of the duplicated frame pairs

In this paper, we proposed an efficient and robust frame duplication detection technique to detect duplicated frames from the video sequence based on the improved Levenshtein distance. At First, we divided the video sequence into small overlapping subsequences and measure the similarity of them by using the improved Levenshtein distance (ILD). Next, the value of ILD is used to detect the duplication forgery frame, in which the higher the value, the lower the similarity between the frame pair. Hence finally, the duplicated frames are located. In the experimental results, the proposed algorithm showed efficacy compared with the state-of-the-art techniques.

The rest of this paper is organized as follows. In related work section, an overview of the related work and contributions in the field of frame duplication forgery detection are provided. Proposed Method section delineates the conceptual and implementation details of the proposed method. The experiments used during performance validation and the obtained results are discussed in Experimental Results section, and the paper is concluded in the last section.

2. Related Work

The frame duplication attacks can be detected from the tampered videos by using the existing digital image forgery detection techniques [10] as the video is a sequence of sequential images in one temporal (time t) and two spatial (x, y) dimensions. However, it may not seem a good idea due to the huge computational complexity obtained rather than the complex scenarios that the videos have such as static scenes [11]. Wang and Farid [12] proposed the first frame duplication forgery detection algorithm by using the spatial and temporal correlations between video frames. A coarse-to-fine comparison manner was used to compare the video subsequences. The high similarities in the temporal correlation coefficients lead to the spatial correlation coefficients comparison. However, their method was unable to detect the frame duplication forgeries in the static scenes and in case of postprocessing attacks such as adding noise on the duplicated frames. Using the previous framework, Yang et al. [6] proposed another two-stage similarity analysis-based method. In the first stage, they extracted the features from each video frame by using the singular value

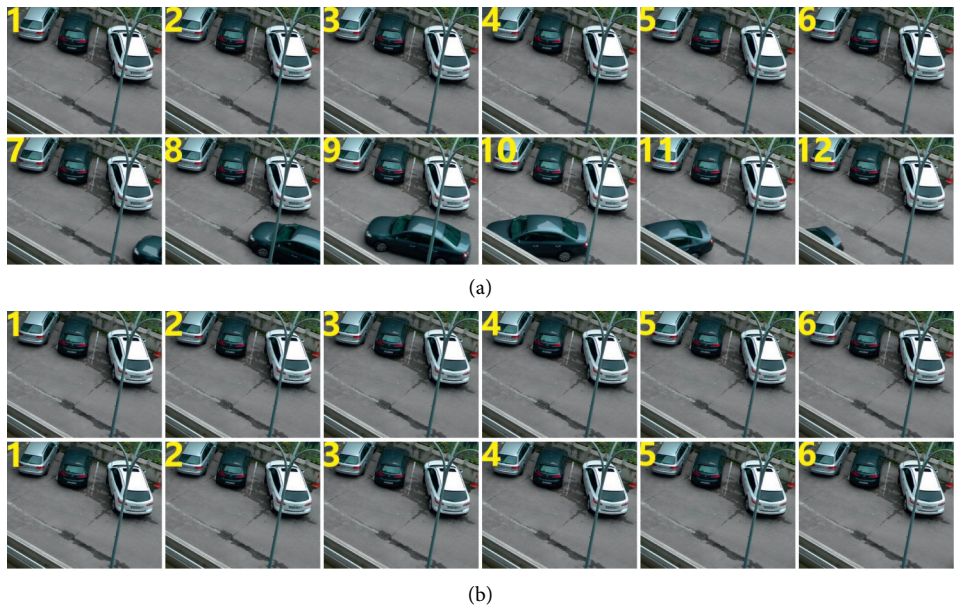


Figure 2: The process of frame duplication attack: an example. (a) The original video sequences. (b) The tampered video sequences.

decomposition (SVD). Then, the Euclidean distance similarity was calculated between the features of reference frame (first frame of the video) and each frame. In the second stage, a random block matching was used to indicate the candidate duplications. However, their method failed to detect the forgeries when frame duplications were done in different order and also when the duplicated frames were less than the window size [1]. Singh et al. [13] divided each video frame into four sub-blocks and then nine features were extracted from each frame. Then, they lexicographically sorted the extracted features to group the most similar frames. Root mean square error (RMSE) was then calculated between the features of adjacent sorted frames to identify the suspicious frames. Then, to detect the frame duplications, the correlation among suspicious frames was performed. Their method failed to detect the forged videos taken by a stationary camera and when duplication was done in different order [1]. Lin and Chang [14] presented an approach for frame duplication detection with four steps: candidate segment selection followed by spatial similarity measurement then frame duplication classification and finally postprocessing. However, many subsequence candidates were selected for the video that results in a significantly high computational time. Li and Huang [15] proposed another frame duplication forgery detection method based on the structural similarity (SSIM) [16]. The similarities between the video subsequences were calculated to find the duplicated frames. However, their method also failed to detect the frame duplication forgeries in the static scenes. D'Amiano et al. [17] proposed an algorithm for frame duplication forgery detection based on the dense-field method with invariant features. They used a suitable video-oriented version of patch-match to limit complexity. Jia et al. [9] proposed a novel approach to detect frame copy-move forgeries based on optical flow (OF), and stable parameters was designed.

The aforementioned methods used predefined fixed global thresholds during the candidates' selection or duplication detection stages. These thresholds may calibrate for a certain condition and may not work for other situations. Moreover, it makes these methods less generalized. Additionally, time complexity is one of the most challenging problems for frame duplication detection algorithms, which increases dramatically with increasing the number of frames within a given video sequence. Furthermore, there is inability to differentiate between the duplicated frame pairs and highly similar frame pairs (misdetected or false positive frame pairs) for the videos with long time static or still scenes.

3. Proposed Method

In interframe forgery (frame duplication), some frames from the video timeline are replaced by a copy from other frames from the same timeline (as shown in Figure 2). In this section, the proposed method for frame duplication forgery detection and localization is introduced in detail. The proposed method includes four stages as shown in Figure 3.

First, the video sequence is divided into small overlapping subsequences; second, similarity measurements based on Levenshtein distance [18] is calculated; third, frame duplication forgery is detected; and fourth, frame duplication forgery is located. To calculate the similarity between the video frames and identify the high similarities of the subsequences, the improved Levenshtein distances for all overlapping subsequences are calculated first. For the experiments, we tampered the video sequences with frame duplication forgery by randomly selecting the location in each video timeline.

3.1. Partition of Video Subsequence. In the experiments, the tampered video sequence $\mathbf{V}$ is first divided into overlapping subsequences ($\mathbf{Seq_i}; i = 1, 2, 3, \ldots, L$), which begin at time $\boldsymbol{\zeta}$.

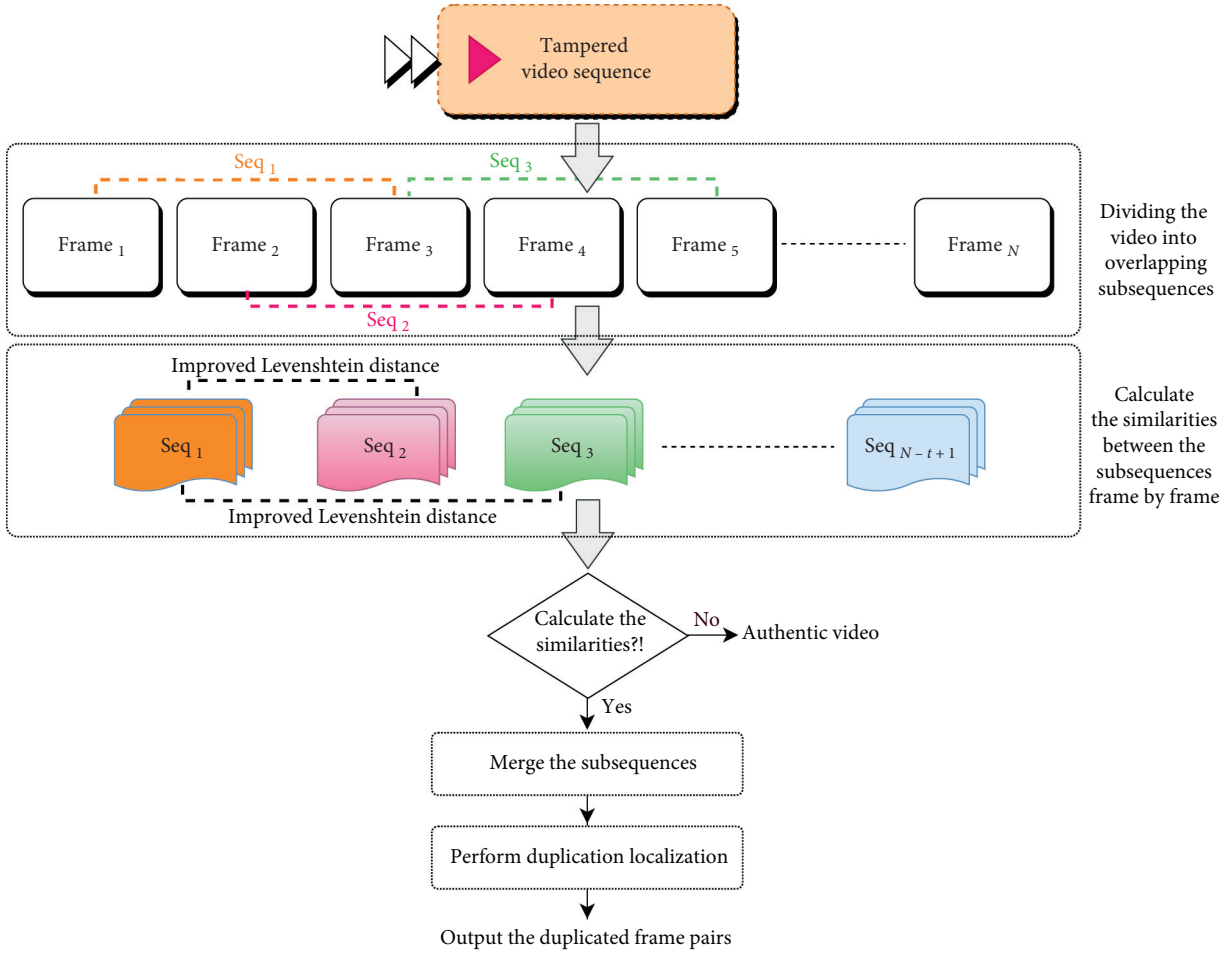


Figure 3: Proposed method steps.

L is the total number of all overlapping subsequences. We assumed that each subsequence length from the overlapping subsequences is (**r**) frames and the length of the test video is (**N**) frames. So, the total number of all overlapping subsequences (**L**) can be given by

$$L = N - r + 1; \quad 1 \le r \le N. \tag{1}$$

Next, we detect the potentially duplicated candidates by calculating the similarities among these subsequences. The similarity of each subsequence has to be calculated with the rest of the other subsequences. The improved Levenshtein distance is adopted and used as a measure of similarities in the proposed method, to measure the similarities between the corresponding frame pairs for each two candidates.

3.2. Similarity Measurements Based on the Improved Levenshtein Distance. The Levenshtein distance is a metric for measuring the similarities between two sets A and B as a simple function of their lengths ($|A|$ and $|B|$) [18]. The generalized Levenshtein distance (GLD) is the most common used measure to compare sets of different edit processes such as insertion, deletion, and substitution of sets elements [19]. The GLD can be obtained from the methods presented in [18, 20]. It shows a distinct tool in some applications as error correction and pattern recognition [21, 22].

Assume that a pair of subsequences $\mathbf{Seq_i}$ and $\mathbf{Seq_j}$ from the tampered video **V** is denoted as $\mathbf{Seq_i} = \mathbf{F_i^1 F_i^2 \ldots F_i^r}$ and $\mathbf{Seq_j} = \mathbf{F_j^1 F_j^2 \ldots F_j^r}$, respectively, where $\mathbf{F_i^m}$ is the **mth** frame of $\mathbf{Seq_i}$ and $\mathbf{F_j^n}$ is the **nth** frame of $\mathbf{Seq_j}$. The length of $\mathbf{Seq_i}$ is given by $|\mathbf{Seq_i}|$. We set up the length of each subsequence to $\mathbf{r = 5}$ and the length of overlap is $\mathbf{r - 1}$.

$\mathbf{T}_{\mathbf{F_i^m},\mathbf{F_j^n}} = \mathbf{T_1 T_2 \ldots T_l}$ is used to show the edit transformation of $\mathbf{F_i^m}$ into $\mathbf{F_j^n}$ which is a sequence of elementary edit processes transforming $\mathbf{F_i^m}$ into $\mathbf{F_j^n}$. Suppose an elementary edit process is (x, y), if a weight function $\boldsymbol{\gamma}$ assigns to $x \longrightarrow y$ a real number (non-negative) $\boldsymbol{\gamma}(x \longrightarrow y)$, the edit transformation weight $\mathbf{T}_{\mathbf{F_i^m},\mathbf{F_j^n}}$ can be computed by $\gamma(\mathbf{T}_{\mathbf{F_i^m},\mathbf{F_j^n}}) = \sum_{\mathbf{i=1}}^{\mathbf{r}} \gamma(\mathbf{T_i})$. Given $\mathbf{F_i^m}$ and $\mathbf{F_j^n}$ are the two frames from **V**, then the generalized Levenshtein distance (GLD) is calculated as in the following equation:

$$GL\ D\left(\mathbf{F_i^m}, \mathbf{F_j^n}\right) = \min\left\{\gamma\left(T_{\mathbf{F_i^m},\mathbf{F_j^n}}\right)\right\}. \tag{2}$$

If γ is a metric over the sequence of elementary edit processes, Marzal and Vidal in [23] defined the GLD as in the following equation:

$$GL\ D\left(\mathbf{F_i^m}, \mathbf{F_j^n}\right) = \min\left\{\mathrm{W}\left(P_{\mathbf{F_i^m},\mathbf{F_j^n}}\right)\right\}, \tag{3}$$

where $P_{\mathbf{F_i^k},\mathbf{F_j^k}}$ is an editing path between $\mathbf{F_i^m}$ and $\mathbf{F_j^n}$ and $W(P_{\mathbf{F_i^m},\mathbf{F_j^n}}) = \sum_{k=1}^{L(P_{\mathbf{F_i^m},\mathbf{F_j^n}})} \gamma(F^m_{i_{k-1}+1\ldots i_k} \longrightarrow F^n_{j_{k-1}+1\ldots j_k})$ is the weight of $P_{\mathbf{F_i^m},\mathbf{F_j^n}}$, which is a set of points or ordered pairs (i_k, j_k), $0 \le k \le \mathrm{L}(P_{\mathbf{F_i^m},\mathbf{F_j^n}}) = l$ satisfying the following conditions:

(1) $0 \le i_k \le |\mathbf{F_i^m}|; 0 \le j_k \le |\mathbf{F_j^n}|; (i_0, j_0) = (0, 0); (i_l, j_l) = (|\mathbf{F_i^m}|, |\mathbf{F_j^n}|)$

(2) $\forall k \ge 1, 0 \le i_k - i_{k-1} \le 1; 0 \le j_k - j_{k-1} \le 1$

(3) $i_k - i_{k-1} + j_k - j_{k-1} \ge 1$

The improved Levenshtein distance similarity (ILD) is normalization for the GLD, and it can be easily computed through GLD. The improved Levenshtein distance between two frames $\mathbf{F_i^m}$ and $\mathbf{F_j^n}$ can be given as follows:

$$\mathrm{ILD}\left(F_i^m, F_j^n\right) = \frac{2.\mathrm{GLD}\left(\mathbf{F_i^m}, \mathbf{F_j^n}\right)}{\alpha.\left(\left|F_i\right| + \left|F_j\right|\right) + \mathrm{GLD}\left(\mathbf{F_i^m}, \mathbf{F_j^n}\right)}, \tag{4}$$

where $|F_i|$ and $|F_j|$ are the length of F_i and F_j, respectively. The final value of the improved Levenshtein distance calculated for two frames is included in the $[0, \infty)$ where 0 means that the two frames are identical (duplicated or replica) whereas any integer number between 1:∞ indicates the number of the different intensities in the corresponding frames.

To illustrate the advantages of ILD, we cut two consecutive frames representing a static scene in an authentic video, as shown in Figure 4. They have a high correlation coefficient that may cause misdetection. For example, the structural similarity (SSIM) between these two frames is calculated and it is equal to 0.9935, which means that if the threshold value in the SSIM-based algorithms [14, 15] is set to be smaller than or equal 0.9935, the detection performance of these frame duplication forgery detection algorithms will decrease dramatically due to the existence of falsely detected frame pairs as duplicated (misdetected frame pairs). Furthermore, they fail to detect the duplication in the static scenes whereas the improved Levenshtein distance between these two authentic frames is equal to 109, which means that there are 109 different pixel intensity values between these frames, and this indicates that these two frames are different and not a duplication from each other.

3.3. Merging Subsequences and Duplication Localization. A helpful distance metric technique significantly improves the performance of localization, clustering, and classification processes [24]. Therefore, the distance metric techniques help algorithms to measure the similarities between the video contents. The tampered video sequence has been divided into small overlapping subsequences to detect frame duplication forgery. In order to form a set of candidate duplicated frames, several duplicated subsequences should be merged to form a complete duplicated sequence. Also, we need to identify which subsequence is original and which subsequence is duplicated (replica).

Due to the small overlapping subsequences, one or more subsequence could match with two or more duplicated subsequences. So, the subsequences with distances equal to 0 between their corresponding frame pairs are selected as a duplicated frame pairs to merge these subsequences in order to form a complete candidate subsequence of duplicated frames. In each subsequence, the similarities between each frame and all other frames of the other subsequences are calculated. Therefore, in this paper, we used the improved Levenshtein distance to calculate the similarities **D[i]** among the corresponding candidate frames (f_i, f_i') as follows:

$$D[i] = \mathrm{ILD}\left(f_i, f_i'\right), \tag{5}$$

where $i = 1, 2, \ldots r$.

Assume that (**S, T**) is a duplicated subsequence, **S** and **T** have the same number of frames (same length), and ($\mathbf{S_i}$, $\mathbf{T_i}$) is a pair of corresponding matched frames. If all the ILD distances $\mathbf{D_i}$ between ($\mathbf{S_i}$, $\mathbf{T_i}$) are equal to 0, then **S** is considered to be the source subsequence and **T** is the duplicated one.

4. Experimental Results

4.1. The Dataset. In the experiments, we selected some test video sequences from the commonly used video test sequences from video trace library (VTL) dataset which is available at http://trace.eas.asu.edu/yuv/index.html. The selected videos are captured with stationary and moving camera modes. The resolution of each one is 352×288 pixels and has the frame rate of 30 fps. Table 1 shows the details of the test tampered videos.

4.2. Performance Evaluation and Analysis. The Precision and Recall rates are used as in equations (6) and (7) to evaluate the detection capability of the proposed method. We also calculate another measure F_1 score that combines both Precision and Recall as shown in equation (8).

$$\mathrm{Precision} = \frac{T_P}{T_P + F_P} \times 100\%, \tag{6}$$

$$\mathrm{Recall} = \frac{T_P}{T_P + F_N} \times 100\%, \tag{7}$$

$$F_1\ \mathrm{score} = 2 \times \frac{Precision \times Recall}{\mathrm{Precision} + \mathrm{Recall}}, \tag{8}$$

where T_P (true positive duplicated frame pairs) represents the number of correctly detected frame pairs as duplicated frames, F_P (false positive duplicated frame pairs) represents the number of falsely detected frame pairs as duplicated frames, and F_N (false negative duplicated frame pairs) represents the number of duplicated frame pairs which are classified as authentic.

To evaluate the performance of our proposed method, we compared our proposed method with Wang and Farid [12] and Li and Huang [15]. The Precision, Recall, and F_1 score rates are calculated for all of the forged videos in the

(a) (b)

Figure 4: A comparison between SSIM and the improved Levenshtein distance for two consecutive frames of an authentic video. (a) Frame no. 2887. (b) Frame no. 2888. (SSIM=0.9935 and improved Levenshtein distance=109).

Table 1: Details of the selected test videos.

Video	Video frames	Video duration (sec)	Frame duplication location
Akiyo	300	10	1~20 are copied to 301~320
Bus	150	05	1~20 are copied to 151~170
Coastguard	300	10	1~20 are copied to 301~320
Container	300	10	1~20 are copied to 301~320
Flower	250	09	1~20 are copied to 251~270
Foreman	300	10	1~20 are copied to 301~320
Hall	300	10	1~20 are copied to 301~320
Mobile	300	10	1~20 are copied to 301~320
Silent	300	10	1~20 are copied to 301~320
Waterfall	260	09	1~20 are copied to 261~280

Table 2: Detection results of the proposed method for the tested video sequences.

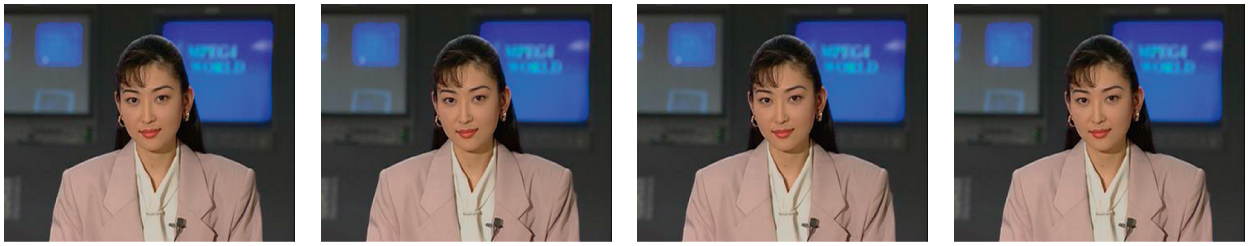
Tampered video	Detection results		
	Proposed method	Wang and Farid [12]	Li and Huang [15]
Akiyo	Original: 1~20 Duplicates: 301~320	Authentic video	Original: 1~20 Duplicates: 301~320, with 192 misdetected frame pairs
Bus	Original: 1~20 Duplicates: 151~170	Original: 1~20 Duplicates: 151~170	Original: 1~20 Duplicates: 151~170
Coastguard	Original: 1~20 Duplicates: 301~320	Authentic video, with 6 misdetected frame pairs	Original: 1~20 Duplicates: 301~320
Container	Original: 1~20 Duplicates: 301~320	Authentic video	Original: 1~20 Duplicates: 301~320, with 39 misdetected frame pairs
Flower	Original: 1~20 Duplicates: 251~270	Original: 1~20 Duplicates: 251~270, with 61 misdetected frame pairs	Original: 1~20 Duplicates: 251~270
Foreman	Original: 1~20 Duplicates: 301~320	Original: 1~20 Duplicates: 301~320, with 17 misdetected frame pairs	Original: 1~20 Duplicates: 301~320
Hall	Original: 1~20 Duplicates: 301~320	Authentic video	Original: 1~20 Duplicates: 301~320
Mobile	Original: 1~20 Duplicates: 301~320	Original: 1~20 Duplicates: 301~320, with 407 misdetected frame pairs	Original: 1~20 Duplicates: 301~320

Table 2: Continued.

Tampered video	Detection results		
	Proposed method	Wang and Farid [12]	Li and Huang [15]
Silent	Original: 1~20 Duplicates: 301~320	Original: 11~18 Duplicates: 311~318	Original: 1~20 Duplicates: 301~320, with 24 misdetected frame pairs
Waterfall	Original: 1~20 Duplicates: 261~280	Original: 186~190 Duplicates: 193~197	Original: 1~20 Duplicates: 261~280

Table 3: Detection capability and location of duplication comparison.

Method	Precision (%)	Recall (%)	F_1 score (%)	Location of duplication
Wang and Farid [12]	28.34	44	34.47	No
Li and Huang [15]	78.88	100	88.19	Yes
Proposed method	99.50	100	99.75	Yes



(a) (b) (c) (d)

Figure 5: Snapshot for the first 4 frames in the test tampered video sequence (Akiyo).

dataset. The higher the Precision as well as the Recall rates and F_1 score are, the better performance will be.

Table 2 shows the detection results of the proposed method for the tested video sequences. It seems that the proposed method is not only able to achieve a high detection of frame duplication forgeries but also accurately locate the duplicated video clips in the video sequences. Table 3 indicates the comparison for the detection capabilities and location of duplication between the proposed method and the methods in [12, 15].

For the test tampered video Akiyo, the frames from 1 : 20 are duplicated in the location from 301 : 320. This video has a static (still) scene as shown in Figure 5, where the first four frames inside that video are visually the same (authentic frames-not duplicated). Figure 6 and Table 2 indicate that the proposed method is able to detect the frame duplication forgeries in the static scenes where the proposed method can correctly detect and locate the frame duplication forgeries (precision rate of 100%). However, the method in Wang and Farid [12] failed and identified this tampered video as an authentic video sequence whereas the method in Li and Huang [15] detected the frame duplication forgeries with low precision rate (9.43%) due to the existence of 192 misdetected frame pairs (false positive duplicated frame pairs). Therefore, the performance of our proposed method is much better than that of the other state-of-the-art methods in [12, 15], as shown in Tables 2 and 3 and Figure 6.

4.3. The Running Time. The comparison between the running time of the proposed method and the methods in Wang and Farid [12] and Li and Huang [15] is shown in Table 4. From that table, we can notice that the method proposed in Wang and Farid [12] has the lowest average time than others. The main reason is that the method proposed by Wang and Farid [12] was unable to locate the location of the duplicated frame pairs, which cost the other methods more time to localize the location of the duplicated frames. However, the method proposed in [12] has the worst detection accuracy than the other algorithms for frame duplication forgery detection (see Figure 6 and Table 2).

All the experiments are conducted on a workstation with Intel Core i7-8750H CPU and 32 GB RAM. We implemented the three methods on MATLAB R2018a.

Generally, the results presented in this paper reveal that the proposed algorithm offers a good performance in comparison with the state-of-the-art techniques. For the future directions, recently, deep learning approaches have been introduced in different fields of detection and identification problems [25–27]. It showed an efficacy and robustness against malicious attacks. Furthermore, copy-move forgery detection (CMFD) algorithms that have been presented for digital images can be used for video frame duplication forgery detection [28–30].

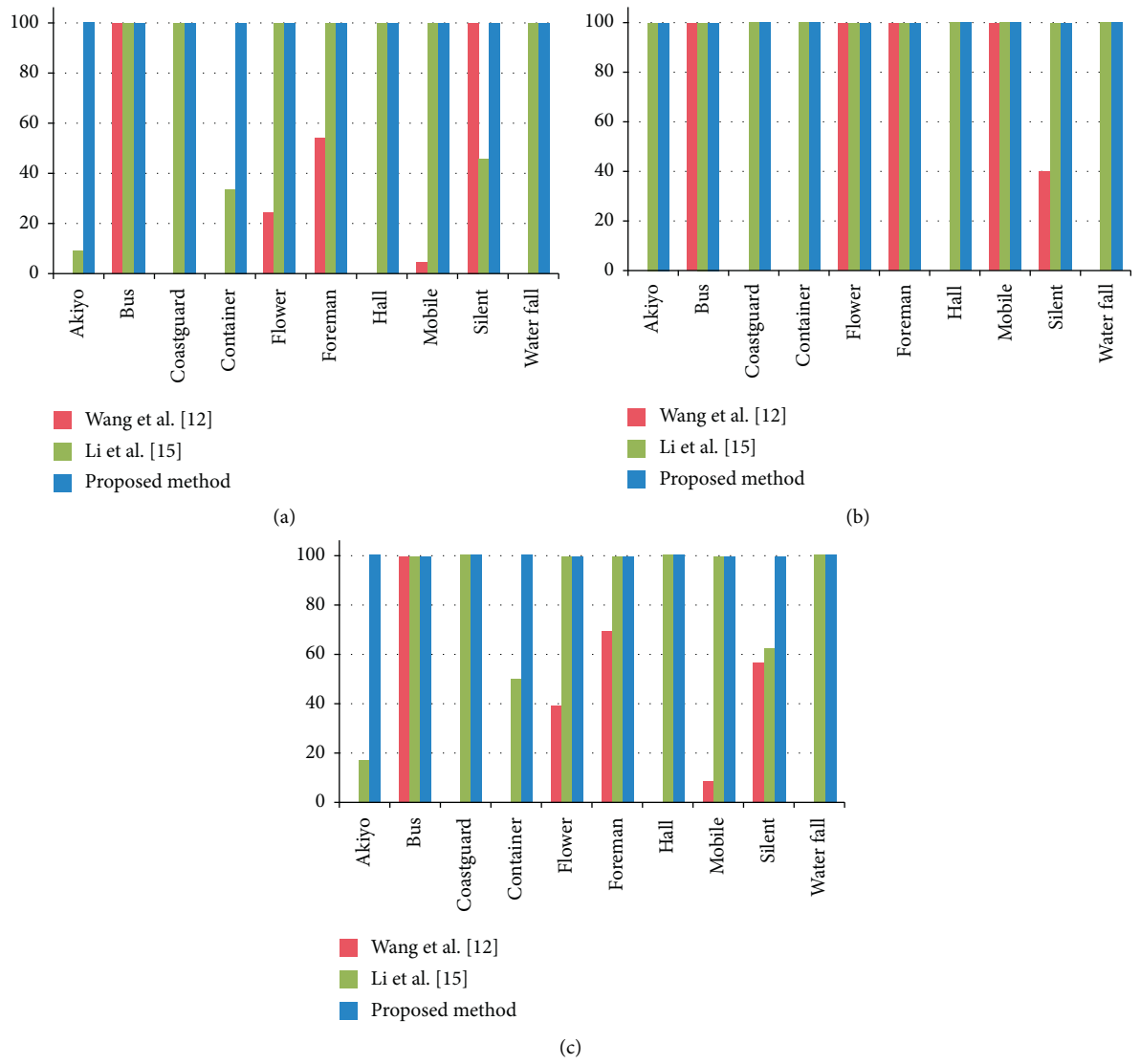


Figure 6: Comparison results with other algorithms. (a) Precision rates. (b) Recall rates. (c) F_1 score.

Table 4: The running time of each tampered video in the dataset.

Tampered video	Video frames (count)	Video duration (sec.)	Average running time (sec.)		
			Proposed method	Wang and Farid [12]	Li and Huang [15]
Akiyo	300	10	16.55	01.10	44.80
Bus	150	05	04.62	02.37	12.47
Coastguard	300	10	16.63	04.46	44.92
Container	300	10	16.40	01.06	45.53
Flower	250	09	11.68	09.12	32.12
Foreman	300	10	16.72	01.94	45.17
Hall	300	10	16.41	01.10	45.67
Mobile	300	10	16.46	14.09	45.66
Silent	300	10	16.17	01.22	45.80
Waterfall	260	09	12.57	0.93	35.01

5. Conclusion

This paper introduces a frame duplication forgery detection and localization approach based on the similarity analysis of the improved Levenshtein distance. The tampered video sequence is first divided into overlapping subsequences. Next, each subsequence has to calculate the similarities with the rest of the other subsequences. The improved

Levenshtein distance is adopted and used as a measure of similarities in this paper. The similarities between all the subsequences are measured to find out the potentially duplicated frame pairs. These duplicated frame pairs are combined together into a complete duplicated sequence, and hence the location of the frame duplication forgeries is located. Extensive experiments are conducted on some tampered videos downloaded from VTL dataset. The results show that the precision of the proposed method can achieve 99.5% which is higher than the state-of-the-art methods. Furthermore, the proposed method is able to locate the exact location of the replica in addition to the detection capabilities of frame duplication forgeries from the static scenes.

Data Availability

No private data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities under grant nos. 2572018BH09 and 2572017PZ10 and Postdoctoral Research Program of Northeast Forestry University under grant no. 203822.

References

[1] K. Sitara and B. M. Mehtre, "Digital video tampering detection: an overview of passive techniques," *Digital Investigation*, vol. 18, pp. 8–22, 2016.

[2] G.-S. Lin, J.-F. Chang, and C.-H. Chuang, "Detecting frame duplication based on spatial and temporal analyses," in *Proceedings of the 2011 6th International Conference on Computer Science & Education (ICCSE)*, pp. 1396–1399, Singapore, August 2011.

[3] R. C. Pandey, S. K. Singh, and K. K. Shukla, "Passive forensics in image and video using noise features: a review," *Digital Investigation*, vol. 19, pp. 1–28, 2016.

[4] W. Wang, X. Jiang, S. Wang, M. Wan, and T. Sun, "Identifying video forgery process using optical flow," in *Proceedings of the 12th International Workshop on Digital-Forensics and Watermarking (IWDW)*, pp. 244–257, Auckland, New Zealand, October 2013.

[5] Q. Dong, G. Yang, and N. Zhu, "A MCEA based passive forensics scheme for detecting frame-based video tampering," *Digital Investigation*, vol. 9, no. 2, pp. 151–159, 2012.

[6] J. Yang, T. Huang, and L. Su, "Using similarity analysis to detect frame duplication forgery in videos," *Multimedia Tools and Applications*, vol. 75, no. 4, pp. 1793–1811, 2016.

[7] Y. Shi, M. Qi, Y. Yi, M. Zhang, and J. Kong, "Object based dual watermarking for video authentication," *Optik*, vol. 124, no. 19, pp. 3827–3834, 2013.

[8] C. Cuevas, E. M. Yáñez, and N. García, "Labeled dataset for integral evaluation of moving object detection algorithms: Lasiesta," *Computer Vision and Image Understanding*, vol. 152, pp. 103–117, 2016.

[9] S. Jia, Z. Xu, H. Wang, C. Feng, and T. Wang, "Coarse-to-fine copy-move forgery detection for video forensics," *IEEE Access*, vol. 6, pp. 25323–25335, 2018.

[10] O. M. Al-Qershi and B. E. Khoo, "Passive detection of copy-move forgery in digital images: state-of-the-art," *Forensic Science International*, vol. 231, no. 1–3, pp. 284–295, 2013.

[11] A. Bovik, "Handbook of image and video processing," *Sensor Review*, vol. 62, no. 4, pp. 4632–4636, 2005.

[12] W. Wang and H. Farid, "Exposing digital forgeries in video by detecting duplication," in *Proceedings of the 9th Workshop on Multimedia & Security*, pp. 35–42, Dallas, TX, USA, September 2007.

[13] V. K. Singh, P. Pant, and R. C. Tripathi, "Detection of frame duplication type of forgery in digital video using sub-block based features," in *Proceedings of the International Conference on Digital Forensics and Cyber Crime*, pp. 29–38, Seoul, Korea, October 2015.

[14] G.-S. Lin and J.-F. Chang, "Detection of frame duplication forgery in videos based on spatial and temporal analysis," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 26, no. 7, p. 1250017, 2012.

[15] F. Li and T. Huang, "Video copy-move forgery detection and localization based on structural similarity," in *Proceedings of the 3rd International Conference on Multimedia Technology (ICMT 2013)*, pp. 63–76, Springer, Berlin, Germany, 2014.

[16] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[17] L. D'Amiano, D. Cozzolino, G. Poggi, and L. Verdoliva, "A patch match-based dense-field algorithm for video copy-move detection and localization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 3, pp. 669–682, 2018.

[18] K. N. S. Behara, A. Bhaskar, and E. Chung, "A novel approach for the structural comparison of origin-destination matrices: Levenshtein distance," *Transportation Research Part C: Emerging Technologies*, vol. 111, pp. 513–530, 2020.

[19] J. Beernaerts, E. Debever, M. Lenoir, B. De Baets, and N. Van de Weghe, "A method based on the Levenshtein distance metric for the comparison of multiple movement patterns described by matrix sequences of different length," *Expert Systems with Applications*, vol. 115, pp. 373–385, 2019.

[20] W. J. Masek and M. S. Paterson, "A faster algorithm computing string edit distances," *Journal of Computer and System Sciences*, vol. 20, no. 1, pp. 18–31, 1980.

[21] J. L. Peterson, "Computer programs for detecting and correcting spelling errors," *Communications of the ACM*, vol. 23, no. 12, pp. 676–687, 1980.

[22] G. Navarro, "A guided tour to approximate string matching," *ACM Computing Surveys*, vol. 33, no. 1, pp. 31–88, 2001.

[23] A. Marzal and E. Vidal, "Computation of normalized edit distance and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 9, pp. 926–932, 1993.

[24] D. A. Adjeroh, M.-C. Lee, and I. King, "A distance measure for video sequences," *Computer Vision and Image Understanding*, vol. 75, no. 1-2, pp. 25–45, 1999.

[25] M. Shafiq, Z. Tian, A. K. Bashir, X. Du, and M. Guizani, "Corrauc: a malicious Bot-Iot traffic detection method in Iot network using machine learning techniques," *IEEE Internet of Things Journal*, vol. 8, no. 5, pp. 3242–3254, 2020.

[26] M. Shafiq, Z. Tian, Y. Sun, X. Du, and M. Guizani, "Selection of effective machine learning algorithm and Bot-IoT attacks

traffic identification for internet of things in smart city," *Future Generation Computer Systems*, vol. 107, pp. 433–442, 2020.

[27] M. Shafiq, Z. Tian, A. K. Bashir, X. Du, and M. Guizani, "IoT malicious traffic identification using wrapper-based feature selection mechanisms," *Computers & Security*, vol. 94, p. 101863, 2020.

[28] M. Emam, Q. Han, and H. Zhang, "Detection of copy-scale-move forgery in digital images using SFOP and MROGH," in *Proceedings of the International Conference of Pioneering Computer Scientists, Engineers and Educators*, pp. 326–334þ, Harbin, China, August 2016.

[29] M. Emam, Q. Han, L. Yu, Y. Zhang, and X. Niu, "A passive technique for detecting copy-move forgery with rotation based on polar complex exponential transform," in *Proceedings of the Seventh International Conference on Digital Image Processing (ICDIP 2015)*, Los Angeles, CA, USA, April 2015.

[30] M. Emam, Q. Han, L. Yu, and H. Zhang, "A keypoint-based region duplication forgery detection algorithm," *IEICE Transactions on Information and Systems*, vol. E99.D, no. 9, pp. 2413–2416, 2016.