Scene Segmentation for Interframe Forgery Identification

Andriani[1], Rimba Whidiana Ciptasari[1] & Hertog Nugroho[2]

[1]School of Computing, Telkom University, Jalan Telekomunikasi. 1,
Kabupaten Bandung 40257, Indonesia
[2]Bandung State of Polytechnic, Jalan Gegerkalong Hilir, Ciwaruga,
Kabupaten Bandung Barat 40559, Indonesia
*E-mail: anisukadji@gmail.com

Abstract. A common type of video forgery is inter-frame forgery, which occurs in the temporal domain, such as frame duplication, frame insertion, and frame deletion. Some existing methods are not effective to detect forgeries in static scenes. This work proposes static and dynamic scene segmentation and performs forgery detection for each scene. Scene segmentation is performed for outlier detection based on changes of optical flow. Various similarity checks are performed to find the correlation for each frame. The experimental results showed that the proposed method is effective in identifying forgeries in various scenes, especially static scenes, compared with existing methods.

Keywords: *inter-frame forgery; optical flow; similarity; static scene; scene segmentation; video forgery.*

1 Introduction

Nowadays, many video editing tools are available to manipulate video. Attackers can use them to tamper with video content and falsify facts. However, the authenticity of video content is difficult to guarantee and requires extra attention if it is used as primary evidence. Video backgrounds contain both static and dynamic scenes. A static scene is characterized by the absence of moving objects or the presence of a static background, whereas a dynamic scene is characterized by moving objects. Although surveillance video often has static scene frames, attackers can exploit it to hide information and the tampering cannot be detected with the human eye.

Video forgeries have two categories: (i) intra-frame forgeries, which occur in the spatial domain, such as the removal of an object from a frame; and (ii) inter-frame forgeries, as shown in Figure 1, which occur in the temporal domain. A common inter-frame forgery consists of frame duplication, insertion, and deletion. This work concerns the identification of inter-frame forgery, which is the easiest type of forgery to carry out. For example, it is easy to delete a person entering a room in a surveillance video by deleting the part of the video where the person appears.

Received September 12th, 2022, Revised February 10th, 2023, Accepted for publication May, 3rd, 2023.

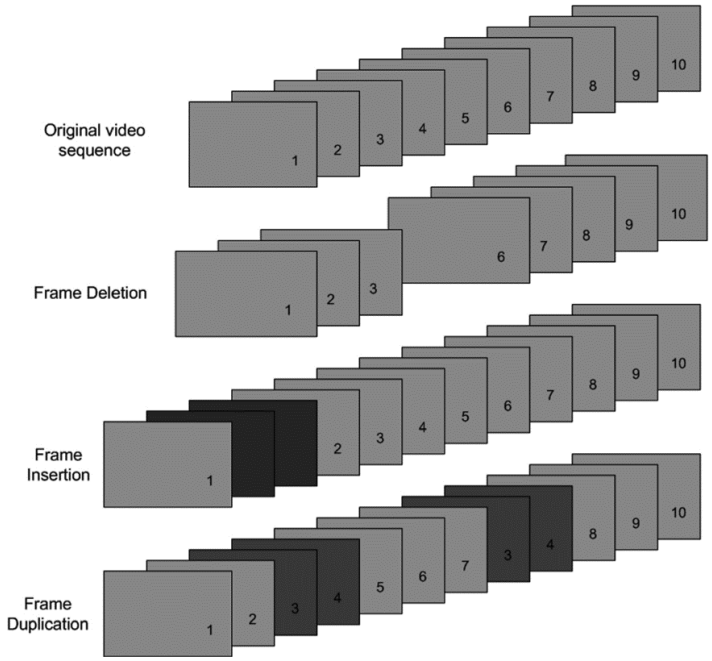

Figure 1 Illustration inter-frame video forgery.

Many researchers have developed forensic systems to expose inter-frame forgery. Fadl, *et al*., in [1], Yang, *et al*., in [2], and Wang, *et al*., in [3], only identified duplication forgery. Fadl, *et al*., in [4], Zheng, *et al.*, in [5], Wang, *et al*., in [6], and Fadl, *et al.*, in [7], identified duplication, frame insertion and deletion forgery. Furthermore, Fadl, *et al.*, in [7], also developed a method to detect shuffling forgery. However, the limitations of these techniques make detecting inter-frame forgery in static scenes difficult.

Our idea is to segment video scenes into static scenes and dynamic scenes, which is our contribution to this field. Based on our observations, static scenes have small changes that are relatively close to zero, while dynamic scenes are characterized by large changes. Outliers in a static scene has small changes in frame deletion, while outliers in a dynamic scene has large changes in tampered videos. Based on the characteristics of each class of scene, our scheme performs forgery identification. It is also capable of detecting forgeries when they occur during class changes. In order to identify various scenes, optical flow is adopted because this method can extract motion features that effectively represent frame conditions.

The rest of this paper is organized as follows. In Section 2, we present several related techniques that are used to explore and enhance forgery detection. Section

3 highlights our main contribution, which is the construction of a scene segmentation algorithm. By developing this algorithm, both static and dynamic segments can be correctly identified. We present the experimental results in Section 4 and conclude this paper in Section 5.

2 Related Works

Fadl, *et al.*, in [1], proposed duplication forgery detection by calculating the standard deviation and similarity between all pairs of feature vectors for subsequential windows and then evaluating the entropy of the Discrete Cosine Transform (DCT) coefficients for each selected residual frame. Yang, *et al.*, in [2], developed an efficient two-stage approach for detecting frame duplication based on similarity analysis.

Wang and Farid, in [3], proposed duplication forgery detection by computing the spatial and temporal correlations among sequential video frames. The method is unsuitable for detecting the duplication forgery in static scenes. Fadl, *et al.*, in [4], computed the differential energy of the residual between frames. The method detects inter-frame forgery (deletion, insertion, and duplication). However, this method requires an original video to identify the forgeries and fails to detect deletion forgery in static scenes. Zheng, *et al.*, in [5], utilized the block-wise brightness variance descriptor (BBVD) for identifying inter-frame deletion and insertion. If forgery occurs, their approach detects inconsistencies in the BBVD ratio at equal time intervals. However, it has a low precision rate for the localization of forgeries.

Wang, *et al*., in [6], used optical flow and anomaly detection to detect inter-frame forgery (i.e., frame deletion, insertion, and duplication). Their method marks discontinuity points in the optical flow variation sequence depending on the type of forgery. However, it fails to detect forgeries in static frames. Fadl, *et al*., in [7], proposed Histogram of Oriented Gradients (HOG) features to detect insertion and deletion forgery. In addition, they calculate the Motion Energy Image (MEI) of edge images to detect duplication and shuffling forgery. However, their method fails to detect frame deletion in static scenes because the frame correlations are high in these scenes.

3 Proposed Method

Some existing works failed to detect forgery in static scenes. We observed that the methods were developed without considering the characteristics of the scenes. A common forgery (shown as a red point in Figure 2) occurs in both static and dynamic scenes. Static scenes show small changes in optical flow, close to zero, while dynamic scenes have larger changes. Based on this observation, we

hypothesized that if we classify the scenes into static and dynamic ones, and then perform forgery detection based on the characteristics of the scene, the process could more effective.

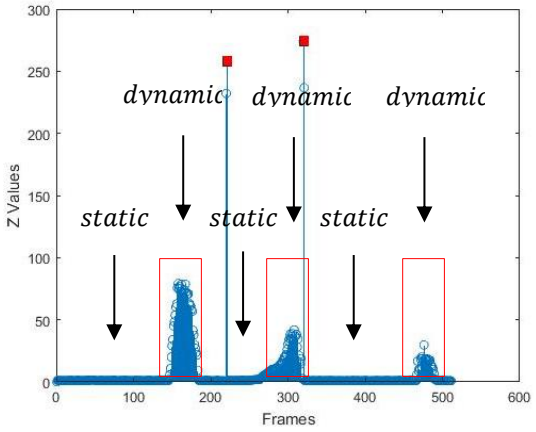

Figure 2 Illustration of static and dynamic scenes in a forgery video.

A block diagram of our system is shown in Figure 3. A questioned video is extracted into sequential frames f (x, y, t), which denote intensity location (x, y) and time t. It is converted into grayscale to reduce color space and resized to 50% of the original image size to reduce computational time. The system consists of two main stages: motion estimation and forgery identification. The following subsections discuss the stages.

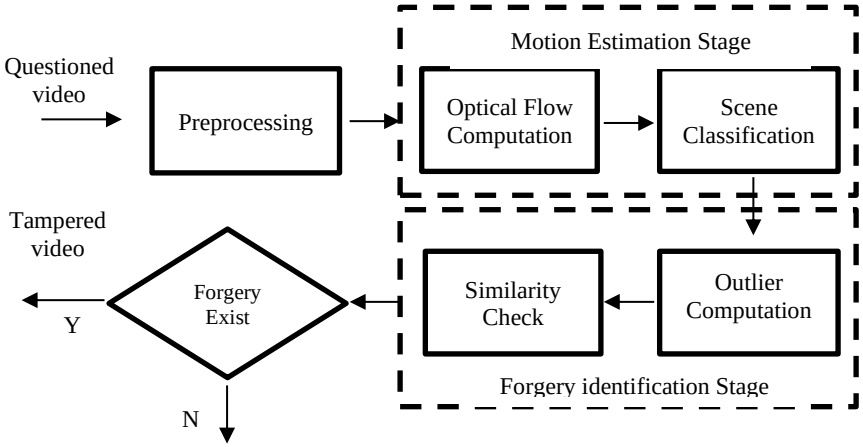

Figure 3 General scheme of video forgery identification.

3.1 Motion Estimation

3.1.1 Optical Flow Computation

The concept of optical flow is widely used for determining the instantaneous velocity of pixel movement of spatially moving objects on the observation imaging plane. To compute the optical flow between two images, the following optical flow constraint equation must be solved:

$$I_x u + I_y v + I_t = 0 \tag{1}$$

where I_x, I_y, and I_t are the spatiotemporal image brightness derivatives, u is the horizontal optical flow, and v is the vertical optical flow. To solve Eq. (1), Lucas-Kanade in [8] performed a weighted least-square fit to combine information from several nearby pixels and assumed a constant velocity in the local neighborhood of the pixel. The method achieves this fit by minimizing Eq. (2):

$$\sum_{x \in \Omega} W^2 [I_x u + I_y v + I_t]^2 \tag{2}$$

where W is a window function for each section Ω.

For frame k and $k + 1$, we calculate the optical flow velocity (u, v) at location (x, y) using magnitude MAG denoted by:

$$MAG = \sqrt{(u^2 + v^2)} \tag{3}$$

where MAG is the magnitude of u and v in Ω. In the velocity of optical flow (u, v), u denotes the horizontal optical flow and v denotes the vertical optical flow. Let frame F_k contain the following magnitudes (Eq. (4)):

$$F_k = \begin{bmatrix} MAG_{1,1} & MAG_{1,2} & \dots & MAG_{1,n} \\ \vdots & \vdots & & \vdots \\ MAG_{m,1} & MAG_{m,2} & \dots & MAG_{m,n} \end{bmatrix} \tag{4}$$

First, compute the difference value of the magnitude of each frame (Eq. (5)):

$$diff_k = |F_k - F_{k+1}|, \ \mathrm{k} \in [1, \mathrm{L} - 1] \tag{5}$$

where L is the number of frames .

$$Z_k = \sum_m \sum_n diff_{m,n,k} \quad , \mathrm{k} \in [1, \mathrm{L} - 1] \tag{6}$$

where Z is the set of sums of the different magnitudes.

3.1.2 Scene Segmentation

Scene segmentation is applied to classify scenes into static and dynamic categories. We define a threshold τ to classify static and dynamic scenes by Eq. (7):

$$\tau_Z = \mu_Z + \alpha\sigma_Z \tag{7}$$

$$\mu_Z = \frac{1}{M}\sum_{k=1}^{M} Z_k \tag{8}$$

$$\sigma_Z = \sqrt{\frac{\sum_{k=1}^{M}(Z_k - \mu_Z)^2}{M}} \tag{9}$$

where μ_Z and σ_Z are mean and standard deviation of the different magnitudes in a frame sequence in a video; α is a tolerance factor; M is the number of different magnitudes Z, and Z is the set of the sums of the different magnitudes, which is expressed in Eq. (6). If the tolerance factor α is larger, static scene segments will have a larger interval than dynamic scene segments. Otherwise, a dynamic scene has a larger interval than a static scene. After that, each scene is detected as static if a frame (difference magnitude) is less than the threshold, otherwise it is dynamic. We discriminate each scene by using the threshold τ_Z in Eq. (7) and each scene is detected as:

$$\text{pred}(Z_k) = \begin{cases} static, & Z_k < \tau_Z \\ dynamic, & Z_k \geq \tau_Z \end{cases} \tag{10}$$

Now, we have $S = \{s_1, s_2, s_3, \ldots, s_N\}$, where S is the set of static and dynamic scenes, and N is the number of scenes. The scene segmentation algorithm used is given in Figure 4.

Pseudocode 1 Procedure of Scene Classification

```
μ_Z = mean of magnitude differences Z (Eq. 7)
σ_Z = standard deviation of magnitude difference Z (Eq. 8)
τ_Z = threshold of scene classification (Eq. 9)
M = number of different magnitudes Z
n = minimum scene segment length (10)
m = difference between adjacent frame index

for k from 1 to M do
    build SP_k = pred(Z_k) (Eq. 10)
end for

for i from 1 to |SP| – 1 do
    build a = {i+1 | SP_i ≠ SP_i+1}
end for

for i from 1 to |a| – 1 do
    m = y – x | x, y ϵ a
    If m ≤ n
       build S = {s(x:y) = s(w) | w = x – 1}
end for

for i from 1 to |S| – 1 do
    build b = {i | S_i ≠ S_i+1}
end for
```

Figure 4 Pseudocode for procedure for scene classification.

In Figure 5, frame duplication forgery (b) has frames with low values (1-70, 110-190, 225-400, 430-450) based on the threshold that represents static scenes. Static scenes have a lower magnitude (MAG) value than dynamic scenes. It denotes two abnormal points (71-110, 400-439). Frame insertion forgery (c) represents a static scene in the ranges 1-150, 183-299, and 322-472. It represents two abnormal points (221-320), where another frame from a different video is taken and inserted at the abnormal point. Frame deletion forgery (d) indicates static scene intervals 1-316 and 338-361 and has only one abnormal point at 140 because the scene is deleted from 140 to 190.

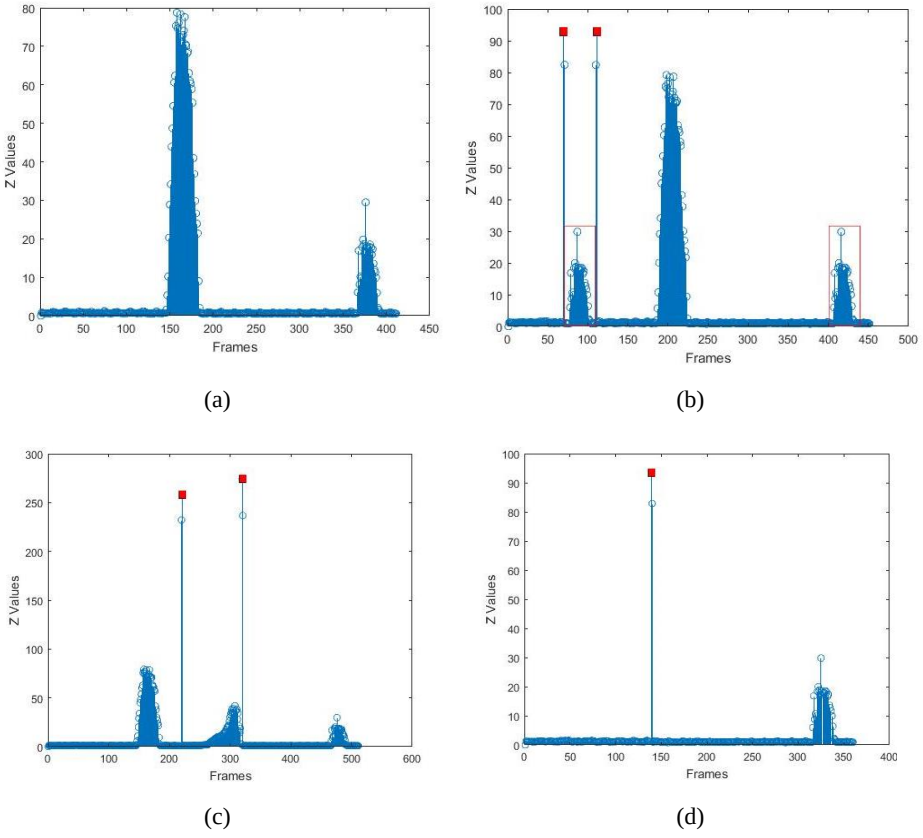

Figure 5 Different of optical flow magnitudes: (a) original video; (b) frame duplication; (c) frame insertion; (d) frame deletion.

3.2 Forgery Identification

3.2.1 Outlier Detection

An outlier in an inter-frame forgery case indicates potential discontinuity points in the optical flow sequence. Hence, we consider an outlier detection scheme to be applied to each scene. Wang, *et al.*, in [6], introduced a variation factor to reveal the relative changes in optical flow sequences. When the range *j* of scene frame fs_j = [1…n], with $fs_j \in s_i$, and *i* = [1…N], then the variation factor vf_i of a dynamic scene is:

$$vf_i = \frac{c \times fs_j}{fs_{j-1}+fs_{j+1}} \tag{11}$$

where fs_{j-1}, fs_j, and fs_{j+1} are adjacent frames in a scene and based on observation we define c = 2. Static scenes tend to have smaller changes in optical flow than dynamic scenes. Therefore, we developed a variation factor in Eq. (11) to represent optical flow changes in static scenes:

$$vf_i = \frac{c \times fs_j}{fs_{j-2}+fs_{j-1}+fs_{j+1}+fs_{j+2}} \tag{12}$$

Next, we calculate the G test statistic of vf in S_i, which requires the mean μ_{vf} and standard deviation σ_{vf} of data set vf.

$$G_i = \frac{|vf_j - \mu_i|}{\sigma_i} \tag{13}$$

where G_i is the G test statistics of vf in S_i; vf is the variation factor per frame in a scene; μ_i and σ_i are the mean and standard deviation respectively for each scene.

However, first we define the forgery threshold *T* as follows: T_{dup} = 4.5; T_{ins} = 7 and T_{del} = 4. If the G_i value is greater than the threshold, accept the point as an outlier. An outlier denotes a forgery candidate in each scene.

$$Outlier = \begin{cases} no\ , & G < T \\ yes, & G \geq T \end{cases} \tag{14}$$

3.2.2 Forgery Classification

To identify forgery types such as duplication, insertion, and deletion forgery, we apply a correlation coefficient to compute the similarity in each outlier scene:

$$C_{ij} = \frac{\sum_{i=1}^{n}(fs_i - \mu_{fs})(fs_j - \mu_{fs})}{\sqrt{\sum_{i=1}^{n}(fs_i - \mu_{fs})^2 \sum_{j=1}^{n}(fs_j - \mu_{fs})^2}} \tag{15}$$

where fs_i and fs_j are two compared frames F_k and μ_{fs} is the mean of the frames in each scene. The detailed pseudocode description is given in Figure 6.

Pseudocode 2 Procedure of Forgery Identification

```
vf_i = variation factor per scene i (Eq. 12 and 13)
μ_vf = mean of variation factor per scene i
σ_vf = standard deviation of variation factor per scene i

for i from 1 to |S| do
    build G_i = {abs (vf_i – μ_vf)/ σ_vf }
end for

stage 1
build DP_ins = {i | G_i > T_ins}
if |DP_ins| ≥ 2
   do foreach pair <i,j> in DP_ins
      C_i−1 = corr2(fs_i, fs_i−1)
      C_i+1 = corr2(fs_i, fs_i+1)
      if (C_i−1 < C_i+1 & C_i−1 < 0.1)
         insertion at i

      C_j−1 = corr2(fs_j, fs_j−1)
      C_j+1 = corr2(fs_j, fs_j+1)
      if (C_i−1 > C_i+1 & C_i+1 > 0.1)
         insertion at j

stage 2
build DP_del = {i | G_i > T_del}
if |DP_del| ≥ 1
   for i from 1 to |DP_del| do
      C_i−1 = corr2(fs_i, fs_i−1)
      C_i+1 = corr2(fs_i, fs_i+1)
      if (C_i−1 < C_i+1 & C_i−1 < 0.65)
         deletion at i

stage 3
build DP_dup = {i | G_i > T_dup}
if |DP_dup| ≥ 2
   do foreach pair <i,j> in DP_dup
      do build S_ij = {fs(k)| k ϵ [i, j]}
         do foreach fs(k) in S_ij and f_l in S_ij'
            C = corr2(fs_k, f_l)
            if (C > 0.98)
               duplication at i,j
```

Figure 6 Pseudocode for the forgery classification procedure.

In the first stage, we define a set of discontinuity points DP_{dup} if G_i is greater than T_{dup} = 4.5. We construct a set subsequence S_{ij} from the pair point in DP_{dup} that will be compared to other frames in S'_{ij}. We compute their similarity and classify a duplication forgery if the result greater than 0.98. We define for frame insertion DP_{ins}, G_i is greater than T_{ins} = 7 and deletion DP_{del} if G_i is greater than

T_{del} = 4.5. The correlation coefficient is computed for each pair point in DP_{ins} and DP_{del} by comparing the current frame fs_i (i.e., outlier point) to the previous frame fs_{i-1} and the current frame fs_i to the next frame fs_{i+1}. The detailed procedure for forgery identification is given in Figure 6.

4 Experimental Results

4.1 Dataset

This research used the dataset from TDTVD (Temporal Domain Tampered Video Dataset) [8], and we selected 68 videos consisting of 8 original videos, 20 frame deletion videos, 20 frame insertion videos, and 20 frame duplication videos. The videos ranged in length from 6 to 18 seconds and had a resolution of 320 x 240 px.

4.2 Parameter Setting

The algorithm was configured as follows: to classify various scenes, we used a tolerance factor α = 1.5; for short interval minimum scene segment length n = 10; the discontinuity point thresholds were T_{dup} = 4.5; T_{ins} = 7; and T_{del} = 4.5; the similarity thresholds were C_{dup} = 0.98; C_{ins} = 1.5; C_{del} = 0.65.

4.3 Performance Analysis

Identification performance was measured using a confusion matrix, i.e., *Precision* (P) and *Recall* (R), which are given by the following formulas:

$$Precision = \frac{TP}{TP+FP} \tag{15}$$

$$Recall = \frac{TP}{TP+FN} \tag{16}$$

where the true positive (TP) rates (i.e., forged is detected as forged with the correct position), false positives (FP) (i.e., authentic is detected as forged), and false negatives (FN) (i.e., forged is detected as authentic). The *F1 score* is the harmonic mean of the scores for *Precision* and *Recall*. It goes from 0% to 100%, and a greater *F1 score* indicates a classifier of higher quality. The *F1 score* is derived from the following mathematical definitions of the *Precision* and *Recall* scores:

$$F1\ Score = \frac{2 \times Precision \times Recall}{Precision+Recall} \tag{17}$$

Table 1 demonstrates that our proposed method is effective for identifying inter-frame forgery. In the case of frame deletion, we obtained one false negative because there was no detected outlier, the *Precision* was 100% free of false

positives and *Recall* was 0.9375, meaning our method could identify frame deletion efficiently in various scenes. For frame insertion detection there was one false positive, because an outlier was detected as a frame deletion that was below the insertion threshold (T_{ins}). However, *Recall* was 100% free of false negatives, and *Precision* was 0.95. For frame duplication detection, which operates based on correlation matching on interval outliers, we obtained *Precision* 100% free of false positives and *Recall* 100% free of false negatives. Furthermore, for all forgery cases, we efficiently identified the location of the forgery.

Table 1 Confusion matrix of dataset.

		Predicted Class			
	Video Type	**Original**	**Deletion**	**Insertion**	**Duplication**
Actual Class	Original	100% (7/7)	-	-	-
	Deletion	6.25% (1/16)	93.75% (15/16)	-	-
	Insertion	-	5% (1/20)	95% (19/20)	-
	Duplication	-	-	-	100% (15/15)

To assess the proposed method's frame-level forgery detection (localization) efficiency, we performed experiments with varying numbers of frame duplications ranging from 25 to 164, frame insertions ranging from 30 to 180, and frame deletions ranging from 31 to 211.

4.4 Comparison with Existing Methods

To evaluate the robustness and performance of our system, the proposed method was compared with similar previous methods in Wang, *et al*., in [6], and Fadl, *et al*. in [7], as given in Table 2.

Table 2 Precision, Recall and F1 Score for all forgeries among methods.

Video Type	**Deletion**			**Insertion**			**Duplication**		
	P	**R**	**F1 Score**	**P**	**R**	**F1 Score**	**P**	**R**	**F1 Score**
Wang *et al.* [6]	1	0.562	0.720	1	0.95	0.974	0.857	0.429	0.572
Fadl *et al.* [7]	0.7	0.438	0.539	0.864	0.95	0.905	0.833	1	0.909
Our method	1	0.938	0.968	0.95	1	0.974	1	1	1

From Table 2, the F1 score varied in the range [0.95, 1] for all cases. This proves that the proposed method is efficient for inter-frame forgery identification. In terms of *Precision*, *Recall* and *F1 Score*, the proposed technique can detect forgery in a static scene for deletion forgery better than the other techniques because it discriminates different types of scenes and perform different variation

factors for each scene segment to detect outliers. In the case of duplication and insertion forgery, the *Precision* of all methods was almost free of false positives, while in the duplication forgery case, the *Recall* of Wang, *et al*. [6] was lower than the others because of false negatives in some forged videos with forgery in various scenes. We also observe that the forgeries probably had an outlier value lower than others, hence, Wang, *et al*. [6] failed to detect the forgeries since they defined deletion and insertion forgeries by the largest value. As a result, we perform similarity checks on outliers from different scenes in order to detect correlations between adjacent outliers.

5 Conclusion

We have shown that our proposed method is an efficient method for common inter-frame forgery (frame duplication, insertion, and deletion) detection, varying between 0.94 and 1 of *Precision* and *Recall*. We introduce scene segmentation to detect anomalies in different scenes. The experimental result was satisfactory for identifying forgeries, although sometimes false positives can occur with objects moving and localized as forged frames or false negatives with forgeries in a static scene.

In future work, we will consider detecting not only single forgeries in a video, but also more than one forgery.

References

[1] Fadl, S.M., Han, Q. & Li, Q., *Authentication of Surveillance Videos: Detecting Frame Duplication Based on Residual Frame*, Journal of forensic sciences, **63**, pp. 1099-1109, 2018.

[2] Yang, J., Huang, T. & Su, L., *Using Similarity Analysis to Detect Frame Duplication Forgery in Videos*, Multimedia Tools and Applications, **75**, pp. 1793-1811, 2016.

[3] Wang, W. & Farid, H., *Exposing Digital Forgeries in Video by Detecting Duplication*, In Proceedings of the 9th workshop on Multimedia & security, pp. 35-42, Sep. 2007.

[4] Fadl, S.M., Han, Q. & Li, Q., *Inter-Frame Forgery Detection Based on Differential Energy of Residue*, IET Image Processing, **13**, pp. 522-528, 2019.

[5] Zheng, L., Sun, T. & Shi, Y.Q., *Inter-Frame Video Forgery Detection Based on Block-wise Brightness Variance Descriptor*, in Digital-Forensics and Watermarking: 13th International Workshop, IWDW 2014, Springer International Publishing, pp. 18-30, 2015.

[6] Wang, W., Jiang, X., Wang, S., Wan, M. & Sun, T., *Identifying Video Forgery Process Using Optical Flow*, in Digital-Forensics and

Watermarking: 12th International Workshop, IWDW 2013, Springer, pp. 244-257, 2014.

[7] Fadl, S., Han, Q. & Qiong, L., *Exposing Video Inter-Frame Forgery Via Histogram of Oriented Gradients and Motion Energy Image*, in Multidimensional Systems and Signal Processing, **31**, pp. 1365-1384, 2020.

[8] Panchal, H.D. & Shah, H.B., *Video Tampering Dataset Development in Temporal Domain for Video Forgery Authentication*, *Multimedia Tools and Applications*, **79**, pp. 24553-24577, 2020.

[9] Grubbs, F.E., *Sample Criteria for Testing Outlying Observations*, The Annals of Mathematical Statistics, pp. 27-58, 1950.

[10] Barron, J.L., Fleet, D.J. & Beauchemin, S.S., *Performance of Optical Flow Techniques*, International Journal of Computer Vision, **12**, pp. 43-77, 1994.

[11] Lucas, B.D. & Kanade, T., *An Iterative Image Registration Technique with an Application to Stereo Vision*, in IJCAI: 7th International Joint Conference on Artificial Intelligence, **81**, pp. 674-679, 1981.