sensors

MDPI

Article

Overcome the Brightness and Jitter Noises in Video Inter-Frame Tampering Detection

Han Pu [1], Tianqiang Huang [1,2,3], Bin Weng [1,2,3,*], Feng Ye [1,2,3] and Chenbin Zhao [4]

[1] School of Mathematics and Information, Fujian Normal University, Fuzhou 350007, China; 20112009@bjtu.edu.cn (H.P.); fjhtq@fjnu.edu.cn (T.H.); yefeng@fjnu.edu.cn (F.Y.)
[2] Digital Fujian Institute of Big Data Security Technology, Fujian Normal University, Fuzhou 350007, China
[3] Engineering Technology Research Center for Public Service Big Data Mining and Application of Fujian Province, Fujian Normal University, Fuzhou 350007, China
[4] School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China; 2020102210008@whu.edu.cn
* Correspondence: binweng@fjnu.edu.cn

Abstract: Digital video forensics plays a vital role in judicial forensics, media reports, e-commerce, finance, and public security. Although many methods have been developed, there is currently no efficient solution to real-life videos with illumination noises and jitter noises. To solve this issue, we propose a detection method that adapts to brightness and jitter for video inter-frame forgery. For videos with severe brightness changes, we relax the brightness constancy constraint and adopt intensity normalization to propose a new optical flow algorithm. For videos with large jitter noises, we introduce motion entropy to detect the jitter and extract the stable feature of texture changes fraction for double-checking. Experimental results show that, compared with previous algorithms, the proposed method is more accurate and robust for videos with significant brightness variance or videos with heavy jitter on public benchmark datasets.

Keywords: brightness-adaptive; robust optical flow algorithm; relax the brightness constancy assumption; texture changes fraction feature

check for updates

Citation: Pu, H.; Huang, T.; Weng, B.; Ye, F.; Zhao, C. Overcome the Brightness and Jitter Noises in Video Inter-Frame Tampering Detection. *Sensors* **2021**, *21*, 3953. https://doi.org/10.3390/s21123953

Academic Editors: Yong Ju Jung, Joohyung Lee, Giorgio Fumera and S. H. Shah Newaz

Received: 15 May 2021
Accepted: 4 June 2021
Published: 8 June 2021

1. Introduction

The rapid development and spread of low-cost and easy-to-use video editing software, such as Adobe Premiere, Photoshop, and Lightworks, makes it easier to tamper with digital video without efforts. Inter-frame forgery happens quite often. It includes inserting frames into a video sequence or removing frames from a video sequence [1]. These tampered videos may be indistinguishable to the naked eye. Thus, they may harm judicial forensics, media reports, e-commerce, finance, and public security. Therefore, it is necessary to develop methods to help human eyes identify tampered videos [1].

A considerable amount of effort has been devoted to inter-frame forgery detection. Most of these approaches are based on the successful extraction of some characteristics of the video. For example, some recent works detected tampered video by calculating the optical flow between frames [2–6]. However, this process could be severely interrupted by illumination noises, which invalidates the extraction of optical flow features [7,8]. Besides, jitter noise may also affect correlation consistency between adjacent frames in the video [9,10], causing many false detections.

For the forgery detection of the videos with noises, a few methods have been developed, including low-rank theory for video with blur noise [11], the coarse-to-fine approach under the condition of regular attacks, including additive noise and filtering [12]. However, these works did not consider brightness and jitter noises. Videos with brightness changes and jitter videos are common in real life—e.g., most of the videos are shot by cell phones. Although the motion-adaptive method [13] considered both brightness and jitter noises, it was not suitable for the lowest motion video with minor changes between adjacent frames, which is quite popular. Moreover, these methods also do not consider or validate the effect of multi-tamper.

We propose a novel framework that not only takes into account both brightness and jitter noises, but also considers the lowest motion video. To deal with considerable illumination noise, we introduce the relaxing brightness constancy assumption [14] and develop a linear model to present the physical intensity change. To deal with subtle illumination noises, we introduce intensity normalization [15]. To deal with the false detection caused by video jitter, we propose motion entropy and stable texture changes fraction features of the video for double-checking. In addition, the improved robust optical flow is insensitive to the motion level of the video. Moreover, the texture changes fraction feature can also describe the subtle inter-frame differences of the lowest motion video. Therefore, our method is also suitable for the lowest motion video. Experimental results on three public video databases show that our method can be applied to the videos with brightness variance, the videos with significant jitter, and the lowest motion videos. Furthermore, our approach can not only locate the forgery precisely, but it can also estimate the way of multi-forgery on tampered positions.

The rest of this paper is organized as follows. In Section 2, we briefly introduce the related work for inter-frame forgery detection. In Section 3, we briefly describe the preliminaries in this paper. Section 4 describes the proposed scheme in detail. We provide the evaluation of optical flow computation in Section 5. Experimental results and analysis are presented in Section 6, and we draw a conclusion and discuss future works in Section 7.

2. Related Work

Most of the prior works detected forgeries based on the analysis of correlations between frames, which relies on features extracted from videos. As noises in videos could significantly affect feature extraction and correlation analysis, we classify the existing methods into two categories: methods without considering noises and methods considering noises.

2.1. Methods without Considering Noises

In terms of the type of features, previous methods could be divided into two categories: image-features-based and video-features-based. Methods in the first category usually extracted image features of each frame, such as texture features [9], color characteristics [9,16], histogram features [16], structural features [17], etc. Methods in the second category mainly utilized the impact of tampering on video features, including video encoding characteristic [18–20], double compression [21], motion features such as errors in motion estimation [10], optical flow, predict residual gradients [19], and brightness features such as segmented brightness variance descriptor (BBVD) [2], illumination information [4], etc.

Although these methods have been validated on videos from public data sets, they generally did not consider noises. They could probably generate incorrect results on real-life videos containing various noises. For example, the performance of methods [2,4,19] declines due to illumination noise, and the methods [9,18] are susceptible to jitter noises in real-life videos.

2.2. Methods Considering Noises

To address the issue of feature extraction in the blurry video, Lin et al. [11] adopted low-rank theory to deblur video, fusing multiple fuzzy kernels of keyframes by low-rank decomposition. Jia et al. [12] proposed a video copy-move detection method based on robust optical flow features. Furthermore, they also adopt adaptive or stable parameters to detect the tamper under the condition of regular attacks, including additive noise and filtering. The method [12] is limited or only validated by copy-move forgery. However, our proposed approach applies to all tampering operations, including frame insertion, frame copy-move, frame replication, and frame deletion. Feng et al. [13] adopted a frame deletion detection method based on the motion residuals feature. They embrace the postprocessing forensic tools, including the automatic color equalization (ACE) forensics and mean gradient evaluation, to eliminate the detection interference caused by illumination and jitter noises.

Illumination noises and jitter noises have side effects on the detection result. However, few works take both brightness and jitter noises into account at the same time. Although [13] has considered both noise factors, it does not fit the lowest motion strength video. While our work takes both brightness and jitter into account at the same time, it is also suitable for the lowest motion videos.

3. Preliminaries

3.1. Horn and Schunck (H&S) Method

When a moving object in the three-dimensional world is projected onto a two-dimensional plane, optical flow (OF) is the relative displacement of the pixels of the image pairs [8]. Specifically, the optical flow method uses the information difference between adjacent frames to describe the movement of objects in a three-dimensional world [7]. OF has been widely applied in various scenes, such as object segmentation, target tracking, and video stabilization [22].

Horn and Schunck (H&S) method [8] is a classical OF estimation algorithm, which is based on three major premise assumptions: brightness consistency, the spatial coherence of neighboring pixels, and small motion of the pixel [23]. Given a video sequence, the pixel intensity at the position (x, y) of t-th frame is $I(x, y, t)$, the brightness consistency can be described by the Equation

$$I(x + dx, y + dy, t + dt) = I(x, y, t) \tag{1}$$

where dx and dy correspond to the slight change of the movement over dt, then Equation (1) can be expanded by the first-order Taylor series

$$I(x + dx, y + dy, t + dt) \approx I(x, y, t) + \frac{\partial I}{\partial x}dx + \frac{\partial I}{\partial y}dy + \frac{\partial I}{\partial t}dt \tag{2}$$

Let $I_x = \partial I/\partial x$, $I_y = \partial I/\partial y$, $I_t = \partial I/\partial t$, then I_x, I_y, I_t represents the change rate of the grey value of the pixel along the x, y, and t directions, respectively. Combining Equations (1) and (2), we can get the Equation

$$^{\partial I}/_{\partial x}dx + {}^{\partial I}/_{\partial y}dy + {}^{\partial I}/_{\partial t}dt = I_x dx + I_y dy + I_t dt = 0 \tag{3}$$

According to the definition of speed Equation $u = dx/dt$ and $v = dy/dt$, we obtain

$$I_x u + I_y v + I_t = 0 \tag{4}$$

Equation (4) is the **OF** constraint equation, then we constrain the **OF** calculation problem to the minimum optimization problem of Equation (5), E_d is the sum of the errors under the brightness constancy constraint, and there are two unknown variables: u and v. An equation cannot determine a unique solution, so a new condition E_s needs to be introduced. E_s is the constraint condition for smooth changes of **OF** over the entire image [24], which is shown in the Equation (6).

$$E_d = \iint (I_x u + I_y v + I_t)^2 dxdy \tag{5}$$

$$E_s = \iint \left(|\nabla u|_2 + |\nabla v|^2 \right) dxdy = \iint_\Omega \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial y} \right)^2 + \left(\frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right] dxdy \tag{6}$$

where ∇ represents the gradient operator.

The H&S algorithm converts the **OF** solution to the minimum optimization problem, shown as the following Equation (7). Equation consists of a grayscale change factor E_d and a smooth change factor E_s. The ideal **OF** value E is relatively small, so the corresponding values of the grayscale change of E_d and the speed change E_s are also small, which meets the assumption of constant brightness and small motion, respectively.

$$E = E_d + \lambda E_s = \iint \left[\left(I_x u + I_y v + I_t \right)^2 + \lambda \left(|\nabla u|_2 + |\nabla v|^2 \right) \right] dxdy \tag{7}$$

where ∇ represents the gradient operator and λ represents the smooth factor.

3.2. Robust Optical Flow Algorithm against Brightness Changes

The above classical **OF** calculation is usually incorrect when the image sequence has significant brightness changes, which exist in most real-life videos. Therefore, the **OF** algorithm was enhanced by relaxing brightness consistency assumptions [14].

Gennert et al. [14] relaxed brightness consistency assumptions by the Equation

$$I(x + dx, y + dy, t + dt) = S(x, y, t) I(x, y, t) + T(x, y, t) \tag{8}$$

where $S(x, y, t)$ and $T(x, y, t)$ are constraint parameters for space and time.

Combining Equations (2) and (8), we can obtain

$$\frac{\partial I}{\partial x} dx + \frac{\partial I}{\partial y} dy + \frac{\partial I}{\partial t} dt = (S - 1) I + T \tag{9}$$

Let $sc = \lim_{dt \to 0} (S - 1)/dt$,$tc = \lim_{dt \to 0} T/dt$, we combine (9) and (3) obtain

$$I_x u + I_y v + I_t - scI - tc = 0 \tag{10}$$

The enhanced **OF** is calculated by solving the extreme value problem described by Equation (11). Compared with Equation (7), the enhanced **OF** algorithm is more robust by considering the brightness change.

$$\begin{gathered} \min_{u,v,sc,tc} E = E_d + \lambda_s E_s + \lambda_{sc} E_{sc} + \lambda_{tc} E_{tc} \\ E_d = \iint_\Omega \left(I_x u + I_y v + I_t - scI - tc \right)^2 dxdy \\ E_s = \iint_\Omega \left(|\nabla u|_2 + |\nabla v|^2 \right) dxdy \\ E_{sc} = \iint_\Omega |\nabla sc|^2 dxdy \\ E_{tc} = \iint_\Omega |\nabla tc|^2 dxdy \end{gathered} \tag{11}$$

where $\lambda_s, \lambda_{sc}, \lambda_{tc}$ are smoothing factor, spatial domain constraint parameter, and time-domain constraint parameter, respectively. E_d, E_{sc} are grayscale change factor and smooth change factor, respectively. E_{sc} and E_{tc} are spatial and time-domain constraint parameters, respectively.

4. Method

We propose a novel framework to overcome the brightness and jitter noises in video inter-frame tampering detection. As illustrated in Figure 1, there are three algorithms in this framework. Firstly, Algorithm 1 reduces the impact of illumination changes in the input video sequence by the optical flow information. At the same time, if the motion entropy is more significant than a certain threshold, we detect jittery video by Algorithm 2. Based on the detected tampering points of the above two steps, Algorithm 3 makes the judgment of video tamper finally.

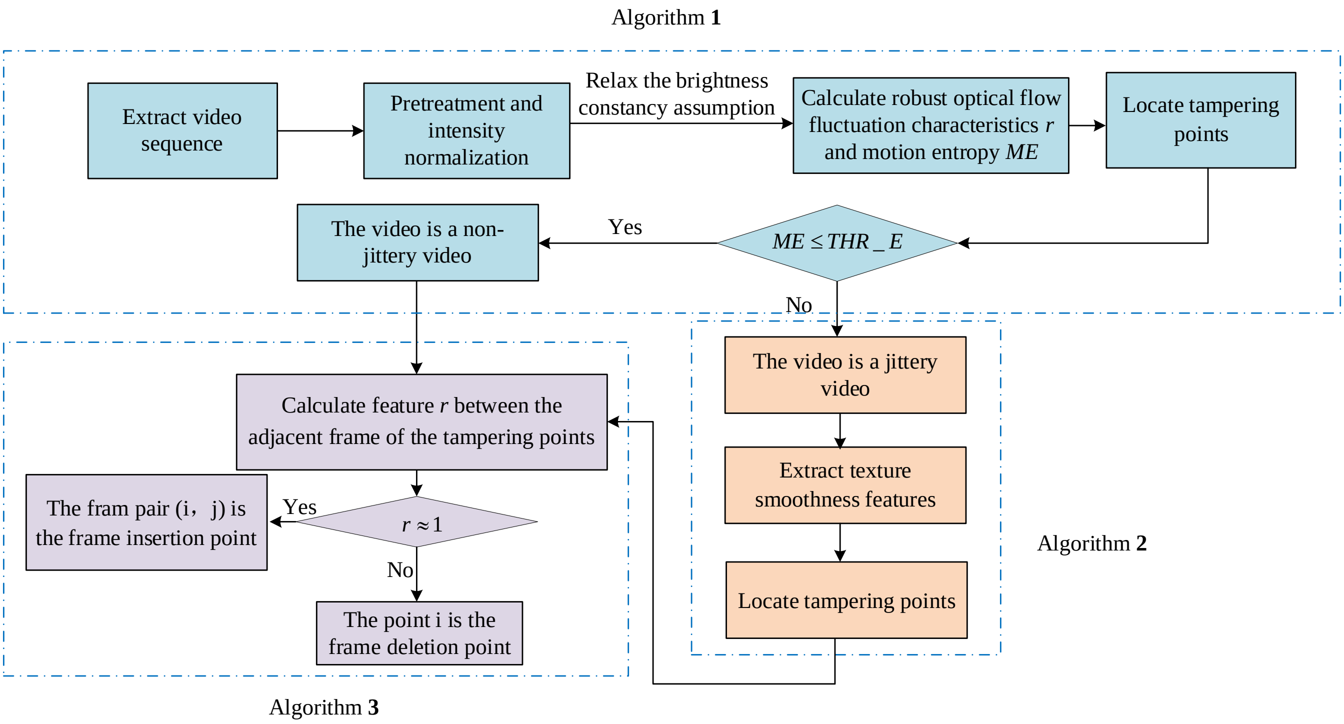

Algorithm **1**
Extract video sequence
Pretreatment and intensity normalization
Relax the brightness constancy assumption
Calculate robust optical flow fluctuation characteristics *r* and motion entropy *ME*
Locate tampering points
The video is a non-jittery video
Yes
ME ≤ *THR* _ *E*
No
The video is a jittery video
Extract texture smoothness features
Locate tampering points
Algorithm **2**
Calculate feature *r* between the adjacent frame of the tampering points
The fram pair (i，j) is the frame insertion point
Yes
r ≈ 1
No
The point i is the frame deletion point
Algorithm **3**

Figure 1. Detection process of the proposed framework.

4.1. Algorithm 1: Reduce the Impact of Illumination Changes

The consistency of the **OF** has been proven to be an efficient tool to check the integrity of video [3,5,6]. Based on the enhanced **OF** algorithm described in the previous section, we design Algorithm 1 to reduce the impact of illumination changes; the main steps are shown as follows.

Step 1: Due to the brightness variations, the intensity of images should be normalized [15] before applying the optical flow method with a digital filter sequence. To cope with the high-frequency noise which affects the **OF** computation, we preprocess the input video by Gaussian filter [25].

Step 2: Based on the enhanced **OF** method described in Section 3 we extract the **OF** fluctuation feature r_i to measure the similarity between adjacent frames of the video by Equation (12)

$$r_i = {}^{sum\_OF_i}/_{avg(sum\_OF_i)} \tag{12}$$

sum_OF_{i} is the **OF** sum of the i-th video frame, which is calculated by Equation (13)

$$\begin{aligned} &sum\_OF_{\mathrm{i}} = \sum_{m=1}^{wid} \sum_{n=1}^{hei} (|u_i(m,n)|) + |v_i(m,n)|) \\ &i = 1, 2, \ldots, N-1 \end{aligned} \tag{13}$$

where wid, hei represent the width and height of the video frame, respectively. N is the video frames number.

The average **OF** sum $avg(sum\_OF_i)$ in a sliding window centered on the i_th frame is calculated by the Equation:

$$avg(sum\_OF_i) = \begin{cases} \frac{sum\_OF_3 + sum\_OF_4}{2} & i = 1 \\ \frac{sum\_OF_{i-1} + sum\_OF_{i+1}}{2} & i \in (1, w+1) \cup (N-w, N) \\ \frac{\sum\limits_{k=1}^{w} sum\_OF_{i+k} + sum\_OF_{i-k}}{2w} & i \in [w+1, N-w] \\ \frac{sum\_OF_{i-1} + sum\_OF_{i-2}}{2} & i = N \end{cases} \tag{14}$$

where $2w$ is the width of the sliding window, sum_OF_3 is the **OF** sum of the third video frame, sum_OF_4 is the **OF** sum of the fourth video frame.

Step 3: Jitter frame pixels have small amplitude movements in the same motion direction [26], which has the consistency of motion direction. The video with consistent motion direction has small motion direction entropy. Therefore, we adopt motion direction entropy ME to perceive the consistency of video motion direction, which can sense video jitter.

ME can be calculated as follows: 1. Use the frame difference method [27] to calculate the binarized motion area. 2. Utilize *Shi-Tomasi* corner calculation method [28] to obtain the corner $c(j)$ on the binarized motion area. 3. Combining the standard deviation $S(\theta)$ of the histogram of the **OF** direction, the motion entropy ME of the video is computed.

$$ME_i = \left(\sum_{j \in c(j)} OF_{ij} \right) / S(\theta) \tag{15}$$

$$\begin{aligned} &ME = std(ME_i) \\ &i = 1, 2, \ldots, N \end{aligned} \tag{16}$$

where OF_{ij} is the **OF** of the corner $c(j)$ in the i_th frame. The $S(\theta)$ is the standard deviation measure of the **OF** direction histogram, which measures the consistency of the direction histogram. ME_i is the motion entropy of the i_th frame and std is the standard deviation of the N video frames.

Step 4: We judge whether the video is tampered with based on the continuity of the video frame feature sequence. THR_R is a threshold selected for the peak point of **OF** fluctuation feature sequence, and C is the variable counter for peak point. THR_E is the threshold selected for ME. If r_i THR_R, the i_th frame is considered to be the suspected tamper point and $C+=1$ is used to count the number of suspected tamper points. When $C \geq 1$, it means the video has suspected tamper points. Under the premise of $C \geq 1$, if the detection result satisfies the condition $ME \leq THR\_E$, which means the video is not jittery. Then we can judge the video as a tampered video directly; if not, it indicates that the video is jittery. Therefore, the video needs to be further detected by Algorithm 2. The suspected tampered position detection process is summarized in Algorithm 1.

Algorithm 1: Reduce the impact of illumination changes

```
Input : video frames$I^{(p)}(1 \leq p \leq N)$, set $THR\_R$ as threshold selected for peak point,
        $THR\_R$ as threshold selected for $ME$.
Output: store position of suspicious tampering point in $S$.
$S = \varnothing, C = 0$ //$C$ is the variable counter for peak point
for $i = 1; i < N; i++$ do
    calculate OF fluctuation feature $r_i$ and motion entropy $ME$
    if $r_i$ $THR\_R$ then
        add $i$into $S$
        $C+=1$
    end if
end for
if $C \geq 1$
    if $ME \leq THR\_E$ then
    (a) return FORGED VIDEO
    (b) store $S, C$
    else run Algorithm 2
    end if
else return ORIGINAL VIDEO
end if
```

4.2. Algorithm 2 Detects Jittery Video

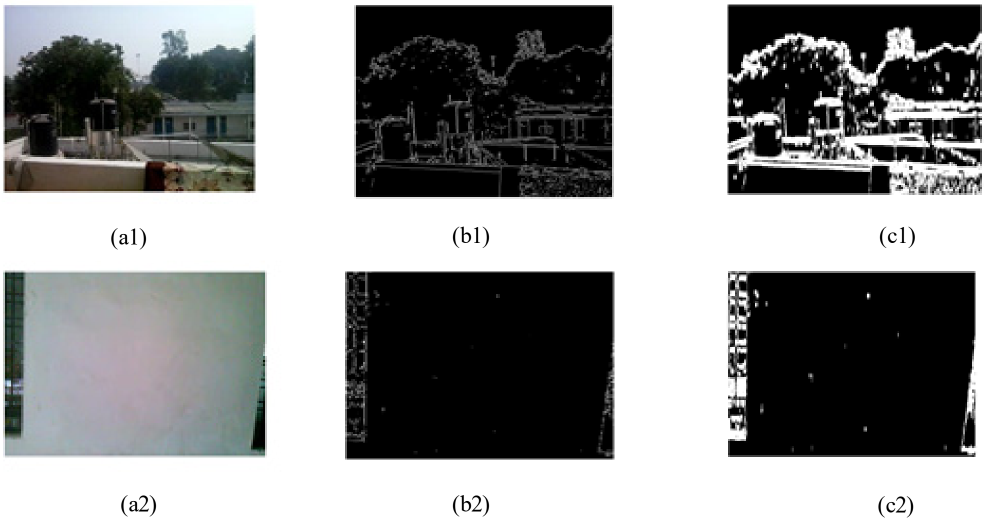
Video jitter refers to a small motion in the same motion direction of the video frame. Since the enhanced **OF** fluctuation feature r in Algorithm 1 is a global motion statistic, only using the feature r is likely to cause leak detection or false detection, especially in the case of severe video jitter. To eliminate the false negative detected point caused by video jitter, we adopt the video texture changes fraction TC to detect the jitter video. The TC feature captures the local details changes of different motion direction of the video frame, which is not captured the characteristics of the same motion direction of the jitter frame, so jittery frames are not identified as tampered frames. The TC feature is calculated by three steps:

Step 1: We compute the gradient structure information of the i_th frame as $\|\Delta I^i\|$. The corresponding binary mask TM_i is obtained by the threshold Th_t for the gradient image $\|\Delta I^i\|$, and the binary mask of the video frame is shown as Figure 2b1,b2.

$$\|\Delta I^i\| = \sqrt{(I_x^i)^2 + (I_y^i)^2} \tag{17}$$

$$TM_i = \begin{cases} 255 & \|\Delta I^i\| \; Th_t \\ 0 & \|\Delta I^i\| \leq Th_t \end{cases} \tag{18}$$

where I_x^i is the partial derivative of the i_th frame in the x-direction, I_y^i is the partial derivative of the i_th frame in the y-direction, and $\|\Delta I^i\|$ means the gradient structure information of the i_th frame.


(a1) (b1) (c1)

(a2) (b2) (c2)

Figure 2. Textured area of video frames. (**a1**,**a2**) are the video frames; after compute the gradient structure information of (**a1**,**a2**), (**b1**,**b2**) are the corresponding binary mask; after perform morphological operations on (**b1**,**b2**), (**c1**,**c2**) are the textures area of video frames).

Step 2: We perform morphological operations on the binary mask TM_i to fill the gaps and remove small areas containing noise, as shown in Figure 2(c1,c2).

$$TM_i = (TM_i \bullet SE) \circ SE \tag{19}$$

where $\bullet$ means a closed operation of morphological operation, $\circ$ means an open operation of morphological operation, and SE is a structural element of open operation and closed operation.

Step 3: We calculate the texture changes fraction $TC(I^i, I^{i+1})$ between TM_i and TM_{i+1} with Equation (20), and $||$ is an absolute value operator. The value of 1 in i_th frame and 0 in $(i+1)\_th$ frame is called the exiting pixel, shown by the arrow at the top of Figure 3, and its statistic is called $Cout$. On the contrary, the value of 0 in i_th frame and 1 in $(i+1)\_th$ frame is called entering pixels, shown by the arrow at the bottom of Figure 3, and its statistic is called Cin. The process of the detection algorithm based on video texture changes fraction TC is shown in Algorithm 2.

$$TC(I^i, I^{i+1}) = |Cin_{\mathrm{i}} - Cout_{\mathrm{i}}| \tag{20}$$

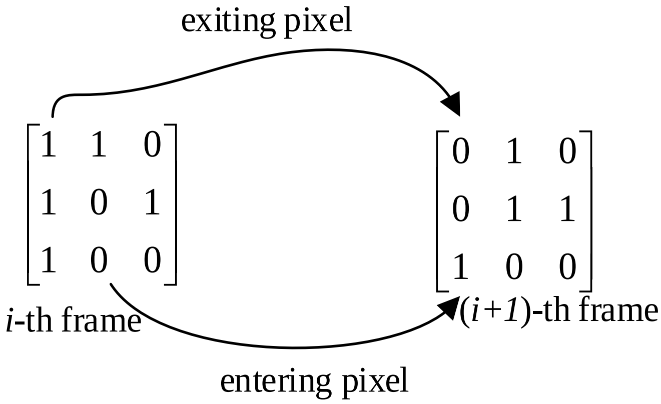


Figure 3. Statistics of video frame texture changes fraction.

Due to the picture continuity of video frames, the content similarity between adjacent frames is substantial, and the value TC is considerably small. If a certain number of frames are inserted or deleted, the video continuity will be destroyed. The larger the value TC, and the more likely the video is to be tampered.

Algorithm 2: Detection algorithm based on video texture changes fraction

```
Input : video frames $I^{(p)}(1 \leq p \leq N)$, set $THR\_R1$ as threshold selected for peak point
Output : store position of suspicious tampering point in $S$
$S = \varnothing, C = 0$ //Reset $S, C$ in Algorithm 1
for $\mathbf{i = 1; i < N; i++}$ do
  calculate video texture changes fraction $TC(I^i, I^{i+1})$
   if $TC(I^i, I^{i+1})$ $THR\_R1$ then
      add $i$
      $C += 1$
end if
end for
if $C \geq 1$ then
(a) return FORGED VIDEO
  (b) store $S, C$
else return ORIGINAL VIDEO
end if
```

4.3. Algorithm 3: Make the Judgement of Video Tamper

The exiting common video tamper operation can cause different tampering point on the extracted video feature sequence. More concretely, the deletion forgery causes a sudden peak in the feature sequence, and the insertion forgery causes two pikes. When I^{i} is a frame forgery point and its previous frame is $I^{(\mathrm{i}-1)}$. At the same time, I^j is another frame forgery point, and its next frame is $I^{(\mathrm{j}+1)}$. If $I^{(\mathrm{i}-1)}$ and $I^{(\mathrm{j}+1)}$ are very similar, then there is a video frame insert clip from I^{i} to $I^{(\mathrm{j}-\mathrm{i}+1)}$. If not, the tamper detection method is a deletion forgery. The process of judgment of video tamper is shown in Algorithm 3.

Algorithm 3: judgment of video tamper

```
Input:suspicious tampering point set in S, the variable counter for peak point C
Output : frame insertion set$S_{insert}$ , frame deletion set $S_{delete}$
$S_{insert} = \emptyset, S_{delete} = \emptyset$
for $i = 1; i < C; i++$ do
  for $j = i + 1; j < C; j++$ do
    if $j > C$:
      add $i$ into $S_{delete}$
else:

        calculate OF fluctuation feature r between frame $S[i] - 1$and $S[j] + 1$
      if $r \approx 1$:
          add $i, j$ into $S_{insert}$
end if
end if
end for
end for
```

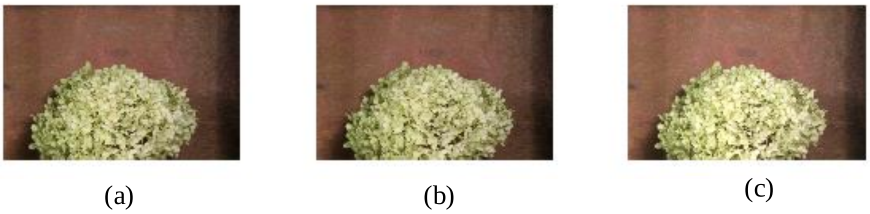
5. Evaluation of Optical Flow Computation

5.1. Experimental Setup

To evaluate the enhanced **OF** algorithm of the proposed detection framework, we perform experiments on the benchmark dataset [29]. The dataset contains various image sequences and the corresponding ground-truth **OF** information, so we can quantify the robustness and accuracy of the enhanced **OF** algorithm. To evaluate the enhanced OF algorithm against dynamic brightness variation, the image I is multiplied by a factor M, and a constant C_1 is added to construct a model of dynamic brightness variation. The specific calculation process is shown in Equation (21). For example, Figure 4a,b show frame10 and frame11 of the Hydrangea sequence group in the dataset, respectively. When $M = 1.1$ and $C_1 = 10$, Figure 4b is changed to Figure 4c. We need to calculate the OF information between Figure 4a,c.

$$I = M * I + C_1 \tag{21}$$

where $M \in [0.9, 1.1], C_1 \in [-10, 10]$.



(a) (b) (c)

Figure 4. Hydrangea image pair. (**a**) frame10, (**b**) frame11, and (**c**) brightness change added on frame11 while $M = 1.1$ and $C_1 = 10$.

We estimate the **OF** information between Figure 4b and c, let u_i^{gt} ,v_i^{gt} represent the real **OF** information, and let u_i^e, v_i^e represent the estimated **OF** information. We evaluate **OF** methods by two measures indicators: the average angular error (AAE) [30] and the end point error (EPE) [31]. The AAE and EPE are used to compare the difference between the ground truth **OF** and the estimated **OF** information. The smaller the values of AAE and EPE, the better the performance of the corresponding **OF** algorithm. We can also visually estimate **OF** algorithm performance by visualization of the flow map. Equation of AAE is shown in Equations (22) and (23).

$$\phi(i) = \arccos\left[\frac{u_i^{gt}u_i^e + v_i^{gt}v_i^e + 1}{\sqrt{\left(u_i^{gt}\right)^2 + \left(v_i^{gt}\right)^2 + 1}\sqrt{\left(u_i^e\right)^2 + \left(v_i^e\right)^2 + 1}}\right] \tag{22}$$

$$\text{AAE} = \frac{1}{N}\sum_{i=1}^{N}\phi(i) \tag{23}$$

5.2. Experimental Results and Analysis

The test results of different **OF** algorithms between frame10 Figure 4a and frame11 (Figure 4c under brightness change) in the Hydrangea sequence group are shown in Figure 5. The description of different approaches and parameter settings used for **OF** evaluation is shown in Table 1. The performance evaluation results of different **OF** algorithms are shown in Table 2. The original image and the ground-truth velocity field are shown in Figure 5a. The flow map and the warped image obtained by the HS algorithm are shown in Figure 5b. The flow map uses different colors and brightness to indicate the size and direction of the estimated **OF**, and the warped image represents frame11 warped to frame10 according to the estimated **OF**. At the same time, it is observed that the estimated flow map in Figure 5b and the ground truth in Figure 5a are significantly different. The error measures of AAE and EPE in Table 2 are also relatively large. It is observed that the HS algorithm is not suitable for the evaluation of the image sequence with dynamic brightness variation.

The evaluation result of the HS+IN (intensity normalization) algorithm is shown in Figure 5c. Compared to the HS algorithm, the values of AAE and EPE of the HS+IN algorithm are significantly reduced. The execution time is not much different, which indicates that the intensity normalization is beneficial to the OF calculation of image sequences with brightness changes.

The evaluation result of the HS+BR (brightness relaxing factor) algorithm is shown in Figure 5d. Compared to the HS algorithm, the values of AAE and EPE of the HS+BR algorithm is greatly increased, which indicates that just introducing the brightness relaxing factor is not beneficial to the OF calculation of image sequences with brightness changes.

Combining IN and BR, we propose the enhanced **OF** algorithm. The evaluation result is shown in Figure 5e, which is very close to the ground-truth velocity field in Figure 5a visually. The warped image is also similar to frame10. The values of AAE and EPE are small, which reaches single digits. The above indicators show that the enhanced **OF** algorithm proposed is suitable for the **OF** calculation of image sequence with brightness changes. We have made a trade-off between computational accuracy and time complexity.

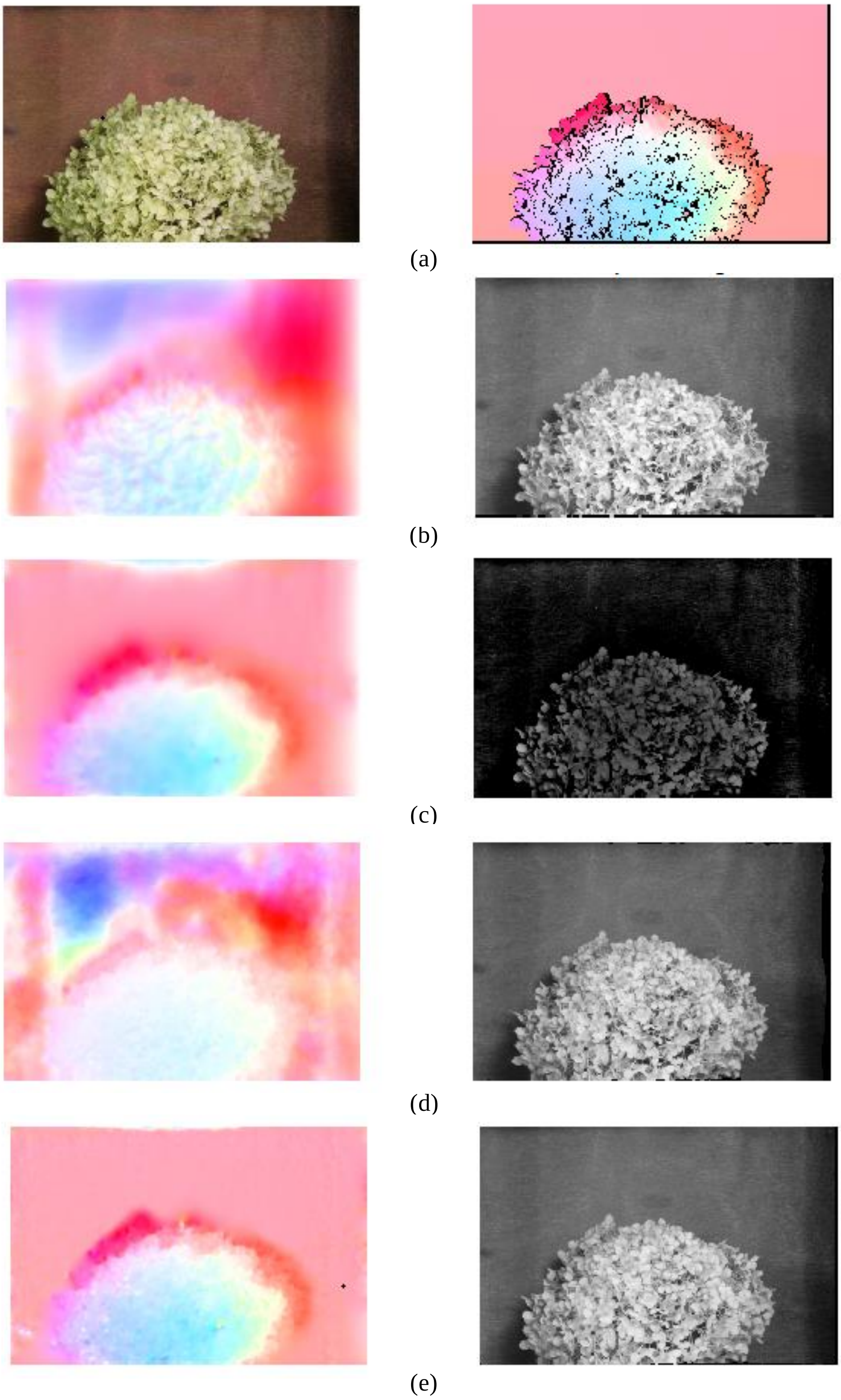


Figure 5. Comparison of different **OF** methods between frame10 and frame11 of Hydrangea images under brightness change (colored). (**a**) Original image and the ground-truth velocity field, **OF** and corresponding warped image of (**b**) 'HS', (**c**) 'HS + IN', (**d**) 'HS + BR', and (**e**) 'the enhanced **OF** algorithm'.

Table 1. Description of different approaches and parameter settings used for **OF** evaluation on Hydrangea images with brightness change.

Approaches	Descriptions and Parameter Settings
HS	Classical H&S method, $\lambda = 1000$.
HS + IN	H&S method with Intensity Normalization, $\lambda = 1000$.
HS + BR	H&S method with Brightness Relaxing factor, $\lambda_s = 10, \lambda_{sc} = 1, \lambda_{tc} = 1$, and d = 0.35.
the enhanced OF algorithm	combine HS+BR and intensity normalization.

Table 2. Error measures of different **OF** methods between frame10 and frame11 of Hydrangea images under brightness change.

Approaches	AAE	Average EPE	Time (s)
HS	13.188	1.350	6.07
HS + IN	7.074	0.776	6.62
HS + BR	28.497	6.631	7.08
Enhanced **OF** algorithm	4.175	0.389	8.06

6. Experimental Results and Analysis

We conduct extensive experiments in diverse and realistic forensic setups to evaluate the performance of the proposed detection framework in this section. The experimental data is introduced first. Then the setup of parameters and evaluation standards are suggested. Finally, we present the experimental results and comparison analysis with four existing state-of-art algorithms to detect accuracy and robustness.

6.1. Experimental Data

To evaluate the detection effect of the proposed method, we performed experiments on three public datasets, namely the SULFA Video Library (The Surrey University Library for Forensic Analysis) [32], the CDNET Video Library (a video database for testing change detection algorithms) [33], and the VFDD Video Library (Video Forgery Detection Database of South China University of Technology Version 1.0) [34], respectively. There are about 200 videos in total. The scenes in the video library are as follows:

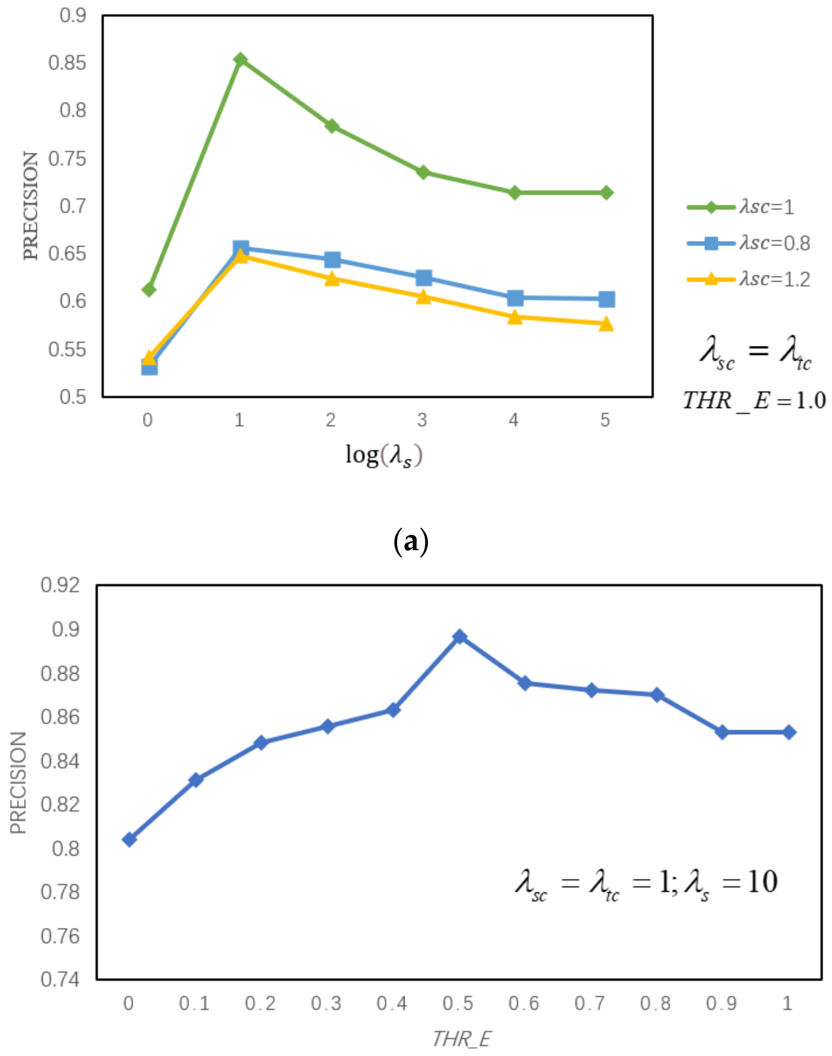
(1) The video library includes videos of different motion levels, including slow motion, medium motion, and high motion.

(2) The video library contains videos of different brightness intensities and different scenes (indoors and outdoors).

(3) The video library includes a variety of mobile phone videos, as well as camera videos, which were taken with or without a tripod.

6.2. Experimental Setup

We download 150 videos with noticeable brightness changes from the video website and adopt the metrics of ACE forensics [35] to determine the brightness changes of videos. We found that these videos have a higher intensity of dynamic brightness changes than the experimental video library. Because the authenticity of the website video is uncertain, it cannot be used as an experimental video. Therefore, we apply the model of dynamic brightness change, which is shown in Equation (21), to simulate the video brightness changes in the real-life environment. We report the precision with respect to λ_{sc}, λ_{tc}, λ_s and THR_E respectively. Based on the results of Figure 6, we observe the effect is best when λ_{sc}= 1, $\lambda_{tc} = 1$, $\lambda_s = 10$ and $THR_E = 0.5$. The values of THR_R and THR_R1 are set according to the Chebyshev inequality adaptively [36], and the corner point $c(i)$ in Algorithm 1 is set to 50.



(**a**)

(**b**)

Figure 6. Ablation study w.r.t. hyperparameters $\lambda_{sc}\Delta\lambda_{tc}, \lambda_s$ and *THR_E*, (**a**) when $\lambda_{sc} = \lambda_{tc}$; *THR_E* $= 1.0$, the precision with the variation of λ_{sc} (**b**)when $\lambda_{sc} = \lambda_{tc} = 1$; $\lambda_s = 10$, the precision with the variation of *THR_E*.

To evaluate the performance of the detection algorithm, we use the error metrics of *precision* and *recall* to analyze the experimental results. The calculation Equations are:

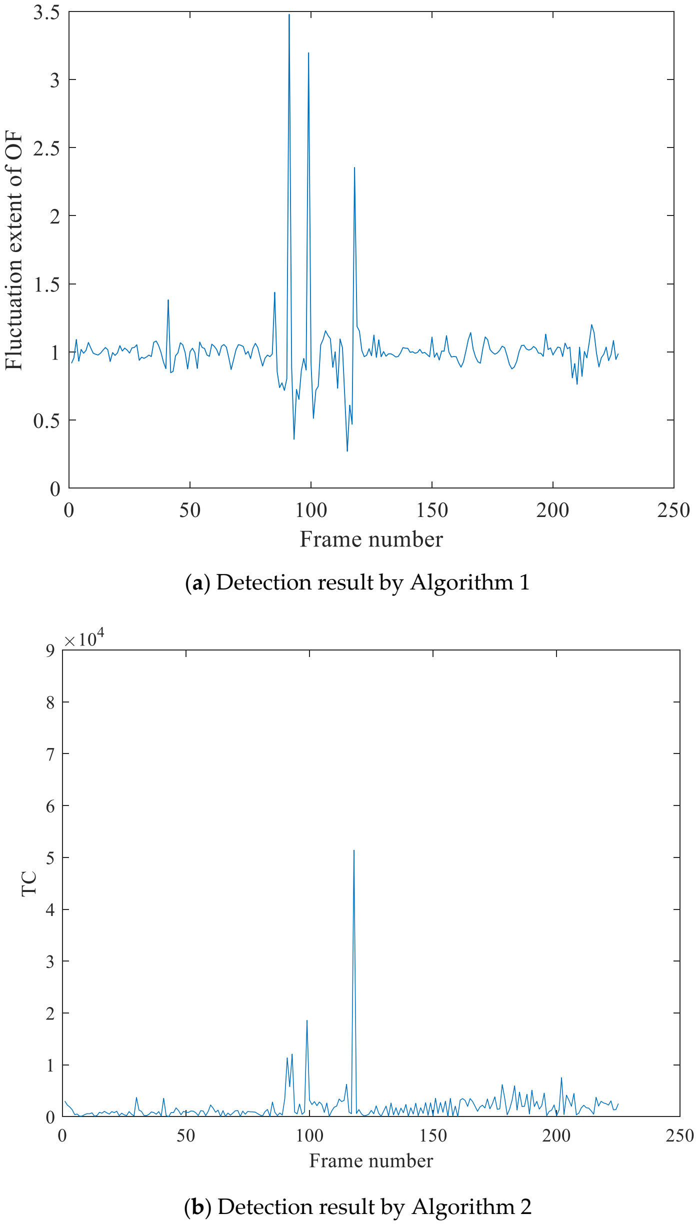
$$precision = \frac{N_c}{N_c + N_f} \tag{24}$$

$$recall = \frac{N_c}{N_c + N_m} \tag{25}$$

where N_c is the number of detected correct points, N_f is the number of detected false points, and N_m is the number of tampered points that were missed.

6.3. Experimental Results

Figure 7 is the detection result of frame deletion forgery for the video with jitter noises and illumination noises. Figure 7a is the experimental results by Algorithm 1, which shows the **OF** fluctuation feature sequence has peaks pair (91, 99, 118). At the same time, the calculated value of motion entropy ME is 0.672, which indicates that the video is jittery. To reduce the side effect of the video jitter, we detect the nervous video by Algorithm 2, which utilizes the texture changes fraction feature TC to detect. The detection result of double-checking is shown as Figure 7b, where the tampering point is 118. At last, we make the judgment of video tamper by Algorithm 3, We can obtain that 118 is frame deletion forgery point, and the peak pair (91, 99) is false detection results.



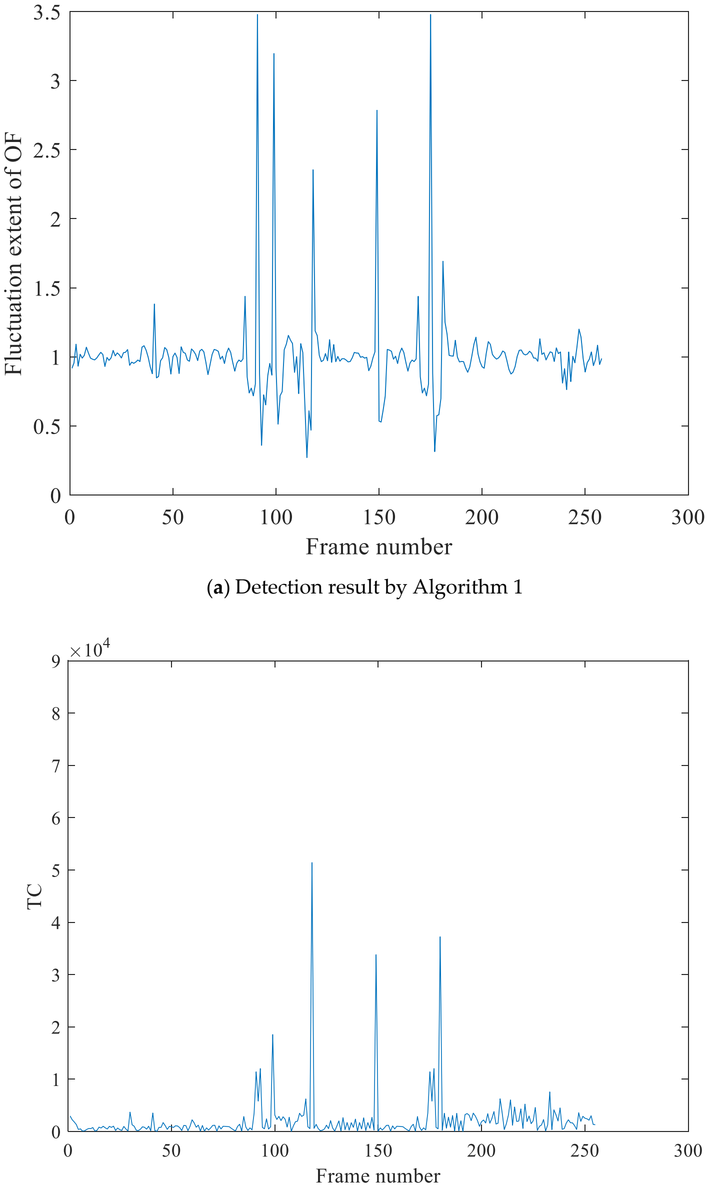
(**a**) Detection result by Algorithm 1

(**b**) Detection result by Algorithm 2

Figure 7. Detection result of frame deletion forgery for the video with jitter noises and illumination noises. In (**a**), Algorithm 1 utilizes the fluctuation extent of **OF** to detect forgery, and the detection results show that the feature sequence has peaks pair (91, 99, 118). At the same time, the motion entropy of **OF** is greater than the selected threshold, which indicates it is a jittery video. Therefore, we detect the video by Algorithm 2, and the detection results in the (**b**) show that frame118 is the tampering point.

Based on the detection result of Figure 7, Figure 8 is the detection result of multiple tampering of the same video. Figure 8a is the experimental results by the Algorithm 1, which shows that the **OF** fluctuation feature sequence has peaks pair (91, 99, 118, 150, 180).

Moreover, the motion entropy *ME* is 0.752, which indicates that the video is jittery. To eliminate the effect of the video jitter, this video is re-tested by Algorithm 2, The re-testing detection result is shown as Figure 8, which locates the tampering points pair at (118, 150, 180), and the peak pair (91, 99) is false detection results. At last, we make the judgment of tamper by Algorithm 3. We can obtain that frame118 is the deletion forgery point, and the point pair (150, 180) is frame insertion forgery point.

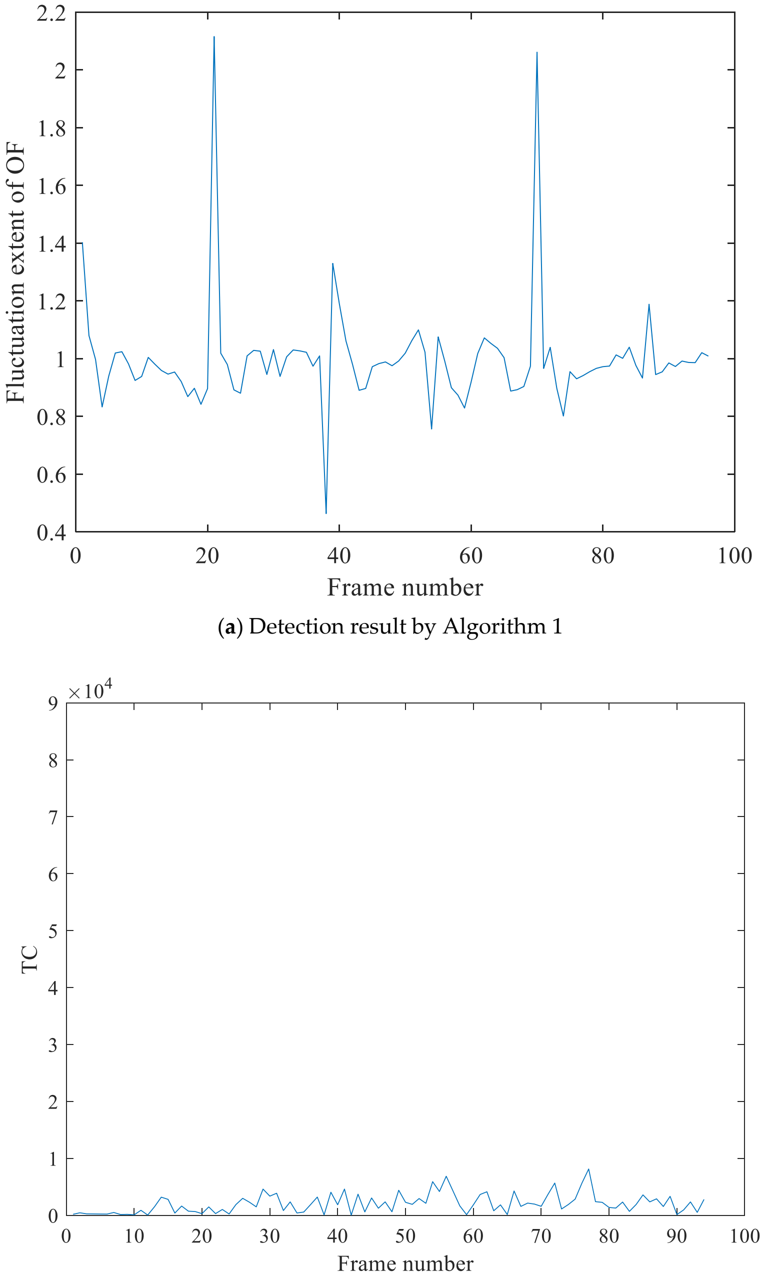


(**a**) Detection result by Algorithm 1

(**b**) Detection result by Algorithm 2 (the jittery video is re-tested by Algorithm 2 and the detection results show that frame pair (118, 150, 180) is the tampering point).

Figure 8. The multi-tamper detection result of jitter video with jitter noises and illumination noises. In (**a**), Algorithm 1 utilized the fluctuation extent of **OF** to detect forgery, and the detection results show that the feature sequence has peaks pair (91, 99, 118, 150, 180). At the same time, the motion entropy of **OF** is greater than the selected threshold, which indicates the video is a jittery video; this video is re-tested using Algorithm 2, and the detection results in (**b**) show that frame pair (118,150,180) is the tampering point.

Figure 9 is the detection result of the untampered video with jitter noises and illumination noises. Figure 9a is the detection result by Algorithm 1, which shown that the **OF** fluctuation feature sequence has a peak pair (22, 70). The motion entropy *ME* is 0.643, which indicates the video is jittery. To eliminate the effect of the video jitter, this video is re-tested by the Algorithm 2, which utilizes the texture changes fraction to detect. The detection result of re-testing is shown as Figure 9b, which indicates that the texture changes fraction sequence has no peaks. Based on the above test results, we judge that the video is original and has not been tampered.



(**a**) Detection result by Algorithm 1

(**b**) Detection result by Algorithm 2

Figure 9. Detection result of the untampered video with jitter noises and illumination noises. In (**a**), Algorithm 1 utilizes the fluctuation extent of **OF** to detect forgery, and the detection results show that the feature sequence has peaks pair (22, 70). At the same time, the motion entropy of **OF** is greater than the selected threshold, which indicates the video is jittery; this video is re-tested using Algorithm 2 and it can be seen that the texture changes fraction sequence has no peaks in (**b**).

Figure 10 shows frame replacement forgery detection result of video with illumination noise. It shows that the feature sequence has peaks pair (51, 93). At the same time, the calculated motion entropy *ME* is 0.453, which indicates that the video is not jittery. Then we judge video tamper, and the **OF** fluctuation feature *r* between frame 50th and 94th is 1.0046, which shows frame pair (50, 94) is very similar. Therefore, the peak pair (51, 93) is the location of video insertion forgery.

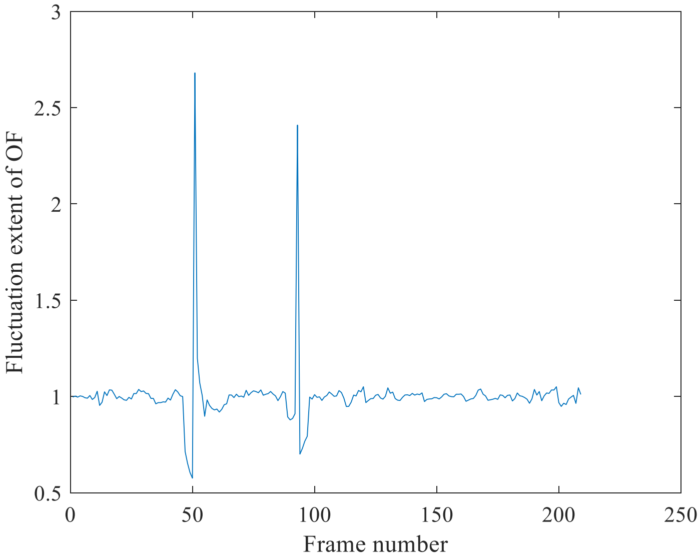


Figure 10. Detection result of frame replacement forgery of video with illumination noise.

Figure 11 shows the detection result of frame deletion forgery of video with illumination noises. It indicates the **OF** fluctuation feature *r* has prominent peaks at frame deletion point 56. Because the motion entropy *ME* is 0.486, which suggests that the video is not jittery. At last, we make the judgment of video tamper and obtain that frame point 56 is the location of video deletion forgery.

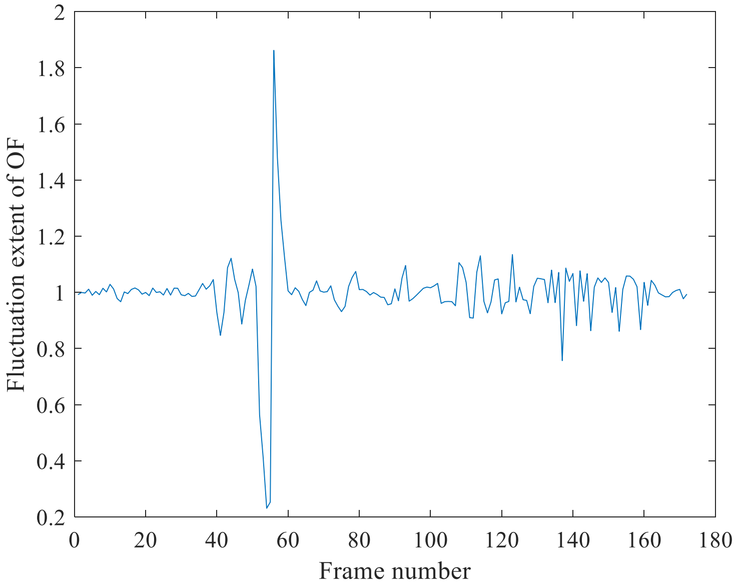


Figure 11. Detection result of frame deletion forgery of video with illumination noise.

Figure 12 is the detection result of video frame copy-move forgery of video with illumination noise. Figure 12 is the detection result by Algorithm 1, which shown that the **OF** fluctuation feature sequence has a peak pair (45, 57). And we calculate the value of motion entropy ME is 0.482, which indicates that the video is not jittery. At last, we make the judgment of video tamper. The **OF** fluctuation feature r between frame 44th and 58th is 0.9844. Therefore, the peak pair (45, 57) is the location of video insertion forgery.

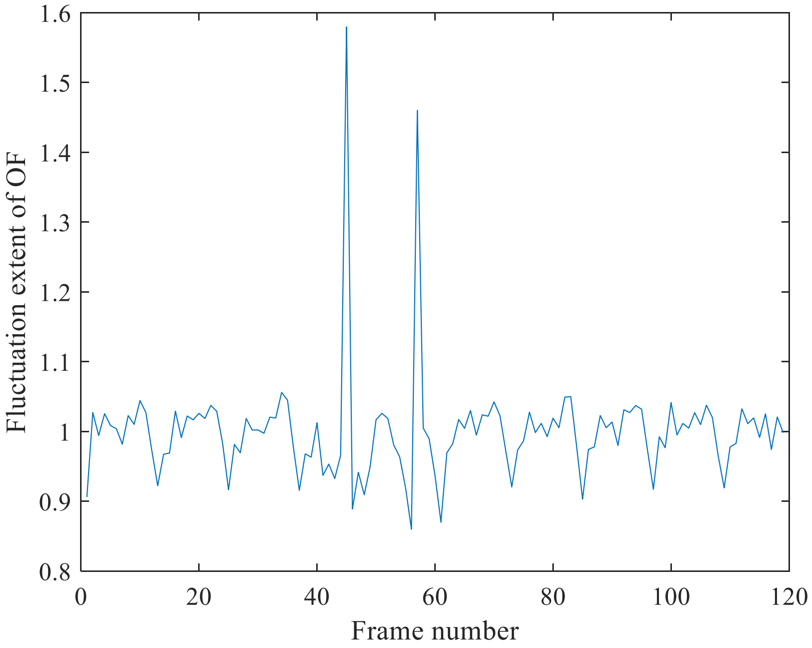


Figure 12. Detection result of frame copy-move forgery of video with illumination noise.

According to the performance evaluation criteria of the proposed algorithm, a comparison is made between the proposed algorithm in the paper and the state-of-the-art different video tamper detection algorithms [3,6,9,37]. Table 3 shows the parameter description of the comparison methods, our proposed method and the comparison methods use the same dataset, and the comparison results are shown in Table 4.

Table 3. Parameter description of comparison methods.

Parameters	Methods				
	Ref. [3]	Ref. [6]	Ref. [9]	Ref. [37]	Proposed
Consider the illumination noise	No	No	Not validated	Not validated	Yes
Consider the jitter noise	Not validated	Not validated	Not validated	Not validated	Yes
Validation by multi-forgery	No	No	No	No	Yes
Forgery detected	Removal/Insertion/copy-move	Copy-move	Removal/insertion/copy-move	Removal/insertion/copy-move	Removal/insertion/copy-move

Table 4. Comparison with state-of-the-art algorithms.

Method	Precision	Recall
Proposed	0.8968	0.8952
Ref. [3]	0.5134	0.5142
Ref. [6]	0.5262	0.5193
Ref. [9]	0.6321	0.6352
Ref. [37]	0.6454	0.6336

As compared to methods reported in [3,6,9,37], the proposed method has high robustness and high accuracy. The results indicate that the proposed method is capable of effective detection and localization of all inter-frame forgeries on videos with illumination noises and jitter noises. In a real-life scenario, the forensic investigator has no control over the parameters of the environment where the video was captured or the parameters used by the video tamper. The forensic investigator must detect in the complete absence of any information regarding the noises, the motion-level, and the forgery operation forms of the captured video. Therefore, the most suitable forgery detection is the one that has practical suitability for the real-life video scenes, such as videos with brightness variance, videos with significant jitter, and the various motion-level videos. Furthermore, our method not only can locate the forgery precisely, but also can estimate the way of multi-forgery on tampered positions.

For [3,6], the detection methods based on **OF** are invalid when there are illumination changes added to the image sequence. Hence, the detection result is not so good. For [9], the detection performance is improved; the main reason is that the Zernike moment feature avoids the effect of brightness intensity. However, experiments prove that its detection performance on the jittery video has decreased significantly, so the detection result is not so good. For [37], the test results are also relatively improved; the main reason is that the multi-channel feature avoids missing detection; however, experimental results show that the performance of this method is not good for the minor frame deletion forgery, so this method is not as stable as the proposed method in our paper.

Prior video tampering detection methods are not suitable for videos with dynamic brightness changes and jittery videos. The detection method [13] based on motion residual can be ideal for the most motion-level video, such as high motion-level, medium motion-level, etc. However, it is not suitable for the slowest motion-level video. The inter-frame difference will decrease as the video motion-level decrease, so the extracted motion residual feature will be weak. However, the relocated I-frame is not affected by the motion level of video, so the relocated I-frame will be defined as the tampered frame mistakenly. Therefore, reference [13] is not suitable for the lowest motion video. Our proposed method utilizes the inconsistencies of features, including the enhanced **OF** and texture changes fraction, to detect tamper in real-life videos. The former feature is insensitive to the motion level of the video. Moreover, the latter feature can also describe the subtle inter-frame differences of the lowest motion video. Therefore, our method is also suitable for the lowest motion video.

To reduce the effect of illumination noises and jitter noises, we utilize a robust optical flow detection method based on relaxing brightness consistency assumption and intensity normalization, which can reduce the influence of significant brightness change and small brightness change, respectively. At the same time, we use motion entropy ME to sense whether the video is jittery and utilize the texture changes fraction TC for double-checking, so the false detection caused by video jitter can be reduced. Experiments prove that the proposed detection method has strong robustness and high accuracy for complex scene video.

7. Conclusions

In this paper, we have proposed a novel detection framework for inter-frame forgery in real-life video with illumination noises and jitter noises. Firstly, for videos with severe brightness changes, we relax brightness constancy constraint and adopt intensity normalization to propose a new optical flow algorithm in Algorithm 1. Secondly, for videos with large jitter noises, we introduce motion entropy to detect the jitter and extract the stable feature of texture changes fraction for double-checking in Algorithm 2. Finally, we make the judgment of video tamper in Algorithm 3. The proposed method was validated by extensive experimentation in diverse and realistic forensic setups. The obtained results indicate that the proposed method is entirely accurate and robust. It can detect video single-forgery or multi-forgeries with an average accuracy of 89%, including frame deletion, frame insertion, frame replacement, and frame copy-move. Furthermore, the proposed method is not sensitive to the jitter noises, illumination noises, or the motion level of the video. In the future, it would be beneficial to explore the suitability of some other real-life video scenes, such as blurred video and still video.

Author Contributions: Methodology, H.P.; Software, H.P.; Validation, H.P.; Investigation, H.P. and T.H.; Data curation, H.P.; Writing—original draft preparation, H.P.; Writing—review and editing, B.W. and H.P.; Visualization, H.P.; Supervision, T.H. and C.Z.; Project administration, T.H.; Funding acquisition, T.H., B.W., and F.Y., All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by National Key R&D Program Special Fund Grant (no. 2018YFC1505805), National Natural Science Foundation of China (nos. 62072106, 61070062), General Project of Natural Science Foundation in Fujian Province (no. 2020J01168), Open project of Fujian Key Laboratory of Severe Weather (no. 2020KFKT04).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Singh, R.D.N. Aggarwal, A. Video content authentication techniques: A comprehensive survey. *Multimed. Syst.* **2017**, *24*, 211–240. [CrossRef]
2. Zheng, L.; Sun, T.; Shi, Y.-Q. Inter-Frame Video Forgery Detection Based on Block-Wise Brightness Variance Descriptor. In Proceedings of the IWDW 2014: Digital-Forensics and Watermarking, Taipei, Taiwan, 1–4 October 2014; pp. 18–30.
3. Kingra, S.; Aggarwal, N.; Singh, R.D. Inter-frame forgery detection in H.264 videos using motion and brightness gradients. *Multimed. Tools Appl.* **2017**, *76*, 25767–25786. [CrossRef]
4. Wu, T.; Huang, T.; Yuan, X. Video Frame Interpolation Tamper Detection Based on Illumination Information. *Comput. Eng.* **2014**, *40*, 235–241.
5. Wang, W.; Jiang, X.; Wang, S.; Wan, M.; Sun, T. Identifying Video Forgery Process Using Optical Flow. In Proceeding of the IWDW 2013: Digital-Forensics and Watermarking, Auckland, New Zealand, 1–4 October 2013; pp. 244–257.
6. Jia, S.; Xu, Z.; Wang, H.; Feng, C.; Wang, T. Coarse-to-fine copy-move forgery detection for video forensics. *IEEE Access* **2018**, *6*, 25323–25335. [CrossRef]
7. Beauchemin, S.S.; Barron, J.L. The computation of optical flow. *ACM Comput. Surv.* **1995**, *27*, 433–466. [CrossRef]
8. Horn, B.; Schunck, B.G. Determining Optical Flow. *Artif. Intell.* **1981**, *17*, 185–203. [CrossRef]
9. Liu, Y.; Huang, T. Exposing video inter-frame forgery by Zernike opponent chromaticity moments and coarseness analysis. *Multimed. Syst.* **2017**, *23*, 223–238. [CrossRef]
10. Sun, T.; Wang, W.; Jiang, X. Exposing video forgeries by detecting MPEG double compression. In Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan, 25–30 March 2012; pp. 1389–1392.
11. Lin, L.; Huang, T.; Pu, H.; Shi, P. Low rank theory-based inter-frame forgery detection for blurry video. *J. Electron. Imaging* **2019**, *28*, 063010.
12. Liao, S.-Y.; Huang, T.-Q. Video copy-move forgery detection and localization based on Tamura texture features. In *International Congress on Image & Signal Processing*; IEEE: New York City, NY, USA, 2014; pp. 864–868.
13. Feng, C.; Xu, Z.; Jia, S.; Zhang, W.; Xu, Y. Motion-adaptive frame deletion detection for digital video forensics. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *27*, 2543–2554. [CrossRef]
14. Gennert, M.A.; Negahdaripour, S. *Relaxing the Brightness Constancy Assumption in Computing Optical Flow*; Massachusetts Inst. of Tech. Cambridge Artificial Intelligence Lab: Cambridge MA, USA, 1987.
15. Kapulla, R.; Hoang, P.; Szijarto, R.; Fokken, J. Parameter sensitivity of optical flow applied to PIV Images. In Proceedings of the Fachtagung “Lasermethoden in der Strömungsmesstechnik”, Ilmenau, Germany, 6–8 September 2011.
16. Zhao, D.-N.; Wang, R.-K.; Lu, Z.-M. Inter-frame passive-blind forgery detection for video shot based on similarity analysis. *Multimed. Tools Appl.* **2018**, *77*, 25389–25408. [CrossRef]
17. Zhang, Z.; Hou, J.; Zhao-Hong, L. Video-frame insertion and deletion detection based on consistency of quotients of MSSIM. *J. Beijing Univ. Posts Telecommun* **2015**, *38*, 84–88.
18. Su, Y.; Zhang, J.; Liu, J. Exposing digital video forgery by detecting motion-compensated edge artifac. In Proceedings of the 2009 International Conference on Computational Intelligence and Software Engineering, Wuhan, China, 11–13 December 2009.
19. Liu, H.; Li, S.; Bian, S. Detecting Frame Deletion in H.264 Video. In Proceedings of the 10th International Conference, ISPEC 2014, Fuzhou, China, 5–8 May 2014; pp. 262–270.
20. Aghamaleki, J.A.; Behrad, A. Malicious inter-frame video tampering detection in MPEG videos using time and spatial domain analysis of quantization effects. *Multimed. Tools Appl.* **2017**, *76*, 20691–20717. [CrossRef]

21. Su, Y.; Xu, J. Detection of double-compression in MPEG-2 videos. In Proceedings of the 2010 2nd International Workshop on Intelligent Systems and Applications, Wuhan, China, 22–23 May 2010; pp. 1–4.
22. Chauhan, A.K.; Krishan, P. Moving object tracking using gaussian mixture model and optical flow. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* **2013**, *3*, 243–246.
23. Nagel, H.-H. On the estimation of optical flow: Relations between different approaches and some new results. *Artif. Intell.* **1987**, *33*, 299–324. [CrossRef]
24. Aisbett, J. Optical flow with an intensity-weighted smoothing. *IEEE Trans. Pattern Anal. Mach. Intell.* **1989**, *11*, 512–522. [CrossRef]
25. Deng, G.; Cahill, L. An adaptive Gaussian filter for noise reduction and edge detection. In Proceedings of the 1993 IEEE Conference Record Nuclear Science Symposium and Medical Imaging Conference, San Francisco, CA, USA, 31 October–6 November 1993; pp. 1615–1619.
26. Feng, C. *Study on Motion-Adaptive Frame Deletion Detection for Digital Video Forensics*; Wuhan University: Wuhan, China, 2015.
27. Singla, N. Motion detection based on frame difference method. *Int. J. Inf. Comput. Technol.* **2014**, *4*, 1559–1565.
28. Ramakrishnan, N.; Wu, M.; Lam, S.-K.; Srikanthan, T. Automated thresholding for low-complexity corner detection. In Proceedings of the 2014 NASA/ESA Conference on Adaptive Hardware and Systems (AHS), Leicester, UK, 14-17 July 2014; pp. 97–103.
29. Baker, S.; Scharstein, D.; Lewis, J.; Roth, S.; Black, M.J.; Szeliski, R. A database and evaluation methodology for optical flow. *Int. J. Comput. Vis.* **2011**, *92*, 1–31. [CrossRef]
30. Finlayson, G.D.; Zakizadeh, R. Reproduction angular error: An improved performance metric for illuminant estimation. *Perception* **2014**, *310*, 1–26.
31. Vint, P.F.; Hinrichs, R.N. Endpoint error in smoothing and differentiating raw kinematic data: An evaluation of four popular methods. *J. Biomech.* **1996**, *29*, 1637–1642. [CrossRef]
32. Qadir, G.; Yahaya, S.; Ho, A.T. Surrey university library for forensic analysis (SULFA) of video content. In Proceedings of the IET Conference on Image Processing, London, UK, 3–4 July 2012; pp. 1–5.
33. Goyette, N.; Jodoin, P.-M.; Porikli, F.; Konrad, J.; Ishwar, P. Changedetection. net: A new change detection benchmark dataset. In Proceedings of the 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, USA, 16–21 June 2012; pp. 1–8.
34. Hu, Y.J.; Salman, A.H.; Wang, Y.F.; Liu, B.B.; Li, M. Construction and Evaluation of Video Forgery Detection Database. *J. S. China Univ. Technol.* **2017**, *45*, 57–64.
35. Jia, S.; Feng, C.; Xu, Z.; Xu, Y.; Wang, T. ACE algorithm in the application of video forensics. In Proceedings of Multimedia, Communication and Computing Application: Proceedings of the 2014 International Conference on Multimedia, Communication and Computing Application (MCCA 2014), Xiamen, China, 16–17 October 2014; p. 177.
36. He, Z. The data flow anomaly detection analysis based on Lip–Chebyshev method. *Comput. Syst. Appl.* **2009**, *18*, 61–64.
37. Huang, T.; Zhang, X.; Huang, W.; Lin, L.; Su, W. A multi-channel approach through fusion of audio for detecting video inter-frame forgery. *Comput. Secur.* **2018**, *77*, 412–426. [CrossRef]