

Research Article

Enhancing Feature Point-Based Video Watermarking against Geometric Attacks with Template

Zhongze Lv ^{1,2}, Hu Guan ¹, Ying Huang ¹, Shuwu Zhang ¹ and Yang Zheng ¹

¹Beijing Engineering Research Center of Digital Content Technology, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

²School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China

Correspondence should be addressed to Ying Huang; ying.huang@ia.ac.cn

Received 26 September 2021; Accepted 3 November 2021; Published 19 November 2021

Academic Editor: Ivan Lee

Copyright © 2021 Zhongze Lv et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As the Internet and communication technologies have developed quickly, the spread and usage of online video content have become easier, which results in major infringement problems. While video watermarking may be a viable solution for digital video content copyright protection, overcoming geometric attacks is a significant challenge. Although feature point-based watermarking algorithms are expected to be very resistant to these attacks, they are sensitive to feature region localization errors, resulting in poor watermark extraction accuracy. To solve this issue, we introduce the template to enhance the location accuracy of feature point-based watermarking. Furthermore, a scene change-based frame allocation method is presented, which arranges the template and the watermark to be embedded into different frames and eliminates their mutual interference, enhancing the performance of the proposed algorithm. According to the experimental results, our algorithm outperforms state-of-the-art methods in terms of robustness against geometric attacks under close imperceptibility.

1. Introduction

Copyright infringement occurs as a result of the rapid development of the Internet [1, 2], which makes it easy to disseminate and utilize digital media assets such as images, videos, and audios. Watermarking, on the other hand, may offer a solution for copyright tracking and verification, which is an algorithm that imperceptibly embeds a watermark containing copyright information into digital content. When copyright complications arise, the watermark is extracted to establish the ownership of the creator. Moreover, the focus of this paper is on video watermarking [3–12], which employs video as its carrier.

Throughout the Internet transmission, several deliberate or unintentional attacks on the watermarked digital video may occur. Common types of attacks include signal processing attacks (adding noise, filtering, transcoding, and so on), frame adjustment attacks (frame insertion, frame dropping, and so on), and geometric attacks (scaling, cropping, and so on). By the way, watermark energy is reduced by signal processing

attacks, frame adjustment attacks change the quantity and relative placement of frames carrying the watermark, and geometric attacks cause the position of the watermark to be desynchronized between embedding and extraction. Most existing video watermarking methods own good robustness against signal processing and frame adjustment attacks, but they are usually vulnerable to geometric attacks.

Because a video can be regarded as a temporally continuous collection of still images, most image watermarking schemes [13–29] are also applicable for video watermarking. Existing image watermarking algorithms offer some ideas for resisting geometric attacks, and video watermarking can use these as references. In addition, feature point-based watermarking [13–21] is a common type of image watermarking against geometric attacks, but it suffers from the watermark location inaccuracy. This paper is aimed at extending feature point-based image watermarking to video watermarking and addressing its existing issue, in order to improve its resistance against geometric attacks.

Feature point-based watermarking extracts feature points from images or frames and uses them to locate nonoverlapping local regions named as feature regions, where the watermark is inserted. The feature regions can be kept as unaltered as feasible before and after geometric attacks by utilizing the invariance of the feature points. The scale information and the spatial coordinate of their associated feature points determine the location and the size of the feature regions; thus, the robustness of the feature regions is largely dependent on the stability of the scale information and the spatial coordinate. However, it is difficult to reappear the picked locations unbiased after suffering geometric attacks, resulting in feature regions locating mistake and a serious decline in the watermark extraction accuracy.

In order to improve the location accuracy, we introduce the template [22–25], which can recover geometrically distorted frames and help feature points in locating the feature regions. To reappear the feature regions from the image which have already recovered by the template, the scale information of the feature points is no longer required, but only the spatial coordinate is required. As a result, the inaccuracy in locating the feature regions can be significantly minimized.

The template is embedded in an image before embedding the watermark, and it is extracted in a possible damaged watermarked image to acquire the affine transform parameters in the extraction procedure. The damaged image is then restored to its original shape, allowing the watermark position to be synchronized between embedding and extraction. However, the watermark and the template are both embedded in the same image at the same time in the existing study, and they will interact with each other. Hence, the algorithm cannot get an ideal performance.

In summary, this paper contributes to the ongoing studies in the following two ways:

- (1) A scene change-based frame allocation strategy is presented. This strategy can effectively arrange different frames to embed the watermark and the template separately, eliminating their mutual interference
- (2) A video watermarking scheme against geometric attacks is designed, combining with the feature points and the template. The template aids feature points in locating the feature regions, decreasing the locating errors and increasing the robustness

After a series of tests on real-world datasets, we find that our approach outperforms the state-of-the-art approaches in terms of robustness, under good imperceptibility.

2. Related Work

The earliest research on video watermarking is in 1994. Matsui et al. [12] consider the video as a collection of sequential images in a specified order and then embed the watermark in these images. A complete video watermarking framework addresses three issues: frame selection for embedding and extraction, embedding region determination, and embedding and extracting scheme design. Selection of frames for embedding and

extraction is a unique task of video watermarking, differing from other digital media watermarking, and it consists of selection of the whole [12] or partial frames (such as I-frames [6, 7], keyframes [8, 9], and scene change frames [10, 11]). The processes of determining embedding regions and designing embedding and extraction schemes are both performed on the selected frames so that the image watermarking algorithms can be used as references. Based on distinct embedding regions, the existing watermarking can be classified into global and local watermarking algorithms. The embedding process of the global watermarking algorithms employs the whole pixels of an image or a frame, making them vulnerable to cropping attacks. The local watermarking algorithms usually select the embedding regions by exploiting the feature point invariance, to make these regions roughly constant before and after attacks. Furthermore, the former has worse imperceptibility than the latter, and the correctness of watermark extraction is related to the precision of locating embedding regions. Quantization [5–7, 26, 27] and spread spectrum [28, 29] are two common types of embedding and extraction schemes. Quantization uses various quantizers to quantize the original carrier data into various index intervals, and the watermark information is extracted based on the index interval to which the quantized data belongs during extraction. Spread spectrum applies the orthogonality of the codebook vector to embed the watermark into the host signal. Quantization is easy to implement with low algorithm complexity, but it is difficult to resist scaling attacks. Spread spectrum has strong robustness against scaling attacks; however, it suffers from the host signal interference.

The feature point-based image watermarking takes some local feature points as the reference points and then uses them to locate some nonoverlapping feature regions, which the watermarks will be embedded into. Tang and Hang [13] utilize the Harris detector to extract feature points and divide the picture into a collection of nonintersect triangles for watermark insertion by Delaunay tessellation. The drawback of this method is that if the feature points retrieved from the original and attacked pictures do not match, the watermark embedding and extraction triangle groups will be different, resulting in the extraction failing. Tang and Hang [14] determine the feature points by using a feature extraction approach called Mexican Hat wavelet scale interaction, and the watermark is embedded in the normalized circular regions centered on these points. Furthermore, several algorithms select the local geometric invariant feature points such as the scale-invariant feature transform (SIFT) [15–19], the Speeded Up Robust Feature (SURF) [20], and the KAZE [21] to locate the feature regions, by using the spatial coordinate and the scale information of these points. Through modifying the pixels in the spatial domain, Lee et al. [16] embed the watermark into the circular patches centered on the chosen SIFT feature points. Zhang et al. [20] present a new watermarking scheme against RST distortion based on SURF and embed the watermark by using the odd-even quantization technique. Liu et al. [21] repeatedly embed the watermarks into the significant bit-planes of the KAZE feature regions, by modifying their histograms. In summary, the general framework of the feature point-based watermarking algorithms follows.

(i) The watermark embedding process

Step 1. Extract the feature points from the image or the frame.

Step 2. Select a particular number of feature points based on certain specified criteria to locate the nonoverlapping feature regions, and their shape is typically square or circular, which is determined as follows:

$$(x - t_1)^2 + (y - t_2)^2 = (ks)^2, \quad (1)$$

where (t_1, t_2) is the spatial coordinate of the selected feature point, s represents the scale information, which is approximately proportional to the scaling factor, and k is a magnification factor to control the radius of the feature regions.

Step 3. Embed the same watermark into these determined feature regions repeatedly, using a specific watermark embedding method. The watermarked image or frame is generated.

(ii) The watermark extraction process

Obtain the feature regions of the watermarked image or frame in the same manner that the embedding process did and then repeatedly extract the watermarks from these regions using the extraction method which corresponds to the embedding method described above. The ownership is proven if the watermark can be identified effectively in at least one region. However, neither (t_1, t_2) nor s can be completely reappeared during extraction, resulting in the feature region desynchronization.

Many watermarking algorithms insert the template in order to recover the geometric attacks, which are mostly manifested in the following structures: Pereira et al. [22, 23] utilize numerous discrete points placed at a specific distance on two straight lines as the template points; Qi et al. [24] use two straight lines as the template; Tokar et al. [25] suggest embedding the square templates in the intermediate frequency transform domain of the images. These methods are resistant to geometric attacks such as scaling and cropping. However, the template and the watermark are both contained in the same image and will interfere with each other. As a consequence, it is difficult for the template-based watermarking algorithms to attain optimal results in terms of imperceptibility and robustness.

3. A Scene Change-Based Frame Allocation Strategy

Because only one image is available as the carrier for the typical template-based image watermarking, the template and the watermark must be embedded into the same image, causing mutual interference. A video, on the other hand, could be seen as a series of images, so it has multiple carriers for embedding. Based on this, we can embed the template and the watermark into different frames, respectively.

In order to embed the template, all of the pixel values of a frame must be changed, which may decrease the imperceptibility. Hence, for embedding the template, it is essential to pick

the frames that are insensitive to human eyes, and the scene change frame fulfils this criterion. The scene change frame is the initial frame of each scene in a video. It changes so quickly during playback that the embedded information is difficult to discover, and using it as a reference point to locate the watermarked frame can significantly increase the extraction efficiency [10]. Based on these advantages, a scene change-based frame allocation strategy shown in Figure 1 is proposed: the scene change frames are chosen to embed the template, and the N frames behind each scene change frame are chosen to embed the watermark, where N is an empirical value.

Moreover, if the current frame is a scene change frame, the correlation coefficient between its histograms and those of the preceding frame will not exceed an empirical threshold. So, we extract the scene change frames based on the correlation coefficient. Denote the correlation coefficient as $d(H(x), H(x+1))$, and then, it is calculated as follows:

$$d(H(x), H(x+1)) = \frac{\text{cov}(H(x), H(x+1))}{\sqrt{D(H(x))D(H(x+1))}}, \quad (2)$$

where $H(x)$ is the histogram of the Y component in the x -th frame, $\text{cov}(H(x), H(x+1))$ is the covariance between $H(x)$ and $H(x+1)$, $D(x)$ is the variance of $H(x)$, and $D(x+1)$ is the variance of $H(x+1)$. If $d(H(x), H(x+1))$ does not exceed a threshold denoted as Thre , then the corresponding frame of $H(x+1)$ is regarded as a scene change frame.

4. The Proposed Video Watermarking Scheme

This section focuses on the video watermarking scheme based on the feature points and the template, which includes the embedding and extraction process.

4.1. Embedding Process. The embedding process is divided into two stages: the template embedding and the watermarking embedding, as shown in Figure 2.

4.1.1. Template Embedding. The template for recovering the linear transform exploits the general property of the Fourier Transform. Define an image as $f(x, y)$, where $0 \leq x < M$, $0 \leq y < N$, and M and N are the width and the height of the image. The DFT is defined as

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \exp\left(-\frac{j2\pi xu}{M} - \frac{j2\pi yv}{N}\right). \quad (3)$$

And its inverse transform is

$$f(x, y) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} F(u, v) \exp\left(\frac{j2\pi xu}{M} + \frac{j2\pi yv}{N}\right). \quad (4)$$

A linear transform in the spatial domain leads to another corresponding linear transform in the DFT domain. That means, if a linear transform G is suffered in the spatial domain as follows:

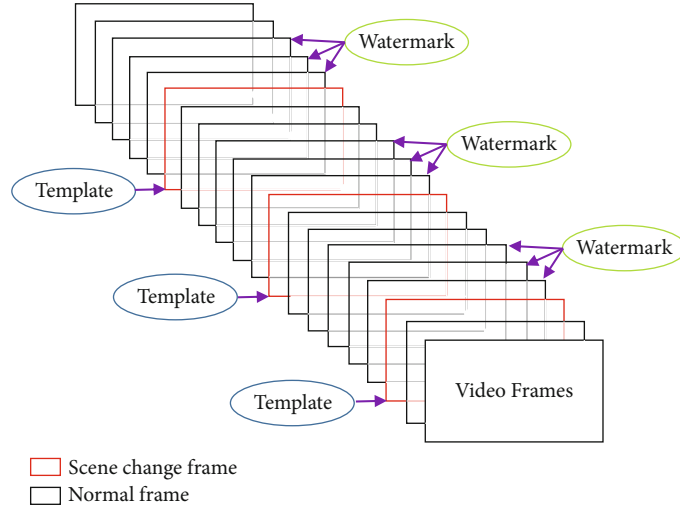


FIGURE 1: The frame allocation strategy based on scene change.

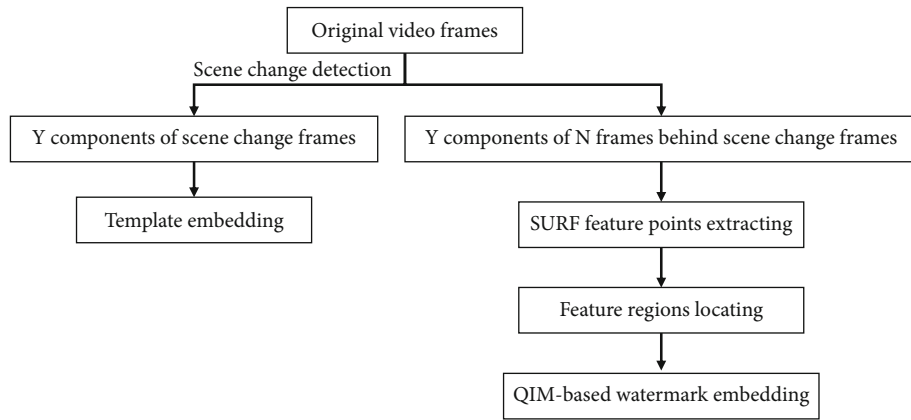


FIGURE 2: The embedding process.

$$\begin{bmatrix} x \\ y \end{bmatrix} \longrightarrow G \begin{bmatrix} x \\ y \end{bmatrix}. \quad (5)$$

Then, the following transform is carried out in the DFT domain, accordingly.

$$\begin{bmatrix} u \\ v \end{bmatrix} \longrightarrow (G^{-1})^T \begin{bmatrix} u \\ v \end{bmatrix}. \quad (6)$$

Thus, by detecting the linear transform of the template in the DFT domain, the corresponding transform in the spatial domain can be deduced.

The template embedding process is shown in the following ways.

Step 1. Perform scene change detection on the host video, according to Equation (2); obtain the Y components of the scene change frames.

Step 2. For each obtained Y component, pad it with zeros to a size of 3000×3000 and then apply the Fast Fourier transform (FFT) to it.

Step 3. Choose two radial lines with an angle of 45 degrees to the coordinate axis in the DFT domain (fixing the angle of 45 degrees can make the extraction process easier because we just need to match the abscissa of the template point); pick seven points in the middle frequency band $[f_1, f_2]$ of each line, and the interval between two adjacent points is equal to 11 pixels (refer to Figure 3); it is necessary to select 7 points at symmetrical positions to obtain real coefficients.

Step 4. For each selected point (x_i, y_i) , embed the template by following equation:

$$\text{MagT}(x_i, y_i) = \text{LocalMean}(x_i, y_i) + \alpha \text{Std}, \quad (7)$$

where $i = 1, \dots, 7$, $\text{MagT}(x_i, y_i)$ is the magnitude of the i -th point containing the template, $\text{LocalMean}(x_i, y_i)$ is the

average of the magnitude of 120 pixels adjacent to the i -th point, and Std is the standard deviation of the whole DFT spectrum.

Step 5. Apply the inverse FFT to the Y component with the template, depadding it to the original size.

4.1.2. Watermark Embedding. We use the QRCode with a size of $W \times W$ as the original watermark, due to its high decoding reliability and strong error correction [11]. The detailed steps of the watermark embedding process are presented below.

Step 1. The scene change frames are extracted from the host video, according to Equation (2), and then, the next N frames of each scene change frame are selected for watermark embedding.

Step 2. Extract the SURF feature points from the Y component of each selected frame, and construct the $R \times R$ feature region with each point as the center; eliminate the points whose corresponding feature regions exceeding the frame boundary; if there are overlapping regions, reserve the point with higher intensity.

Step 3. Select the first three points with the highest intensity among the remaining feature points, and the watermark will be repeatedly embedded in the feature regions corresponding to these points.

Step 4. Segment each selected feature region into $W \times W$ blocks; for each block, calculate the DC coefficient in the discrete cosine transform based on Equation (8).

$$DC_{ij} = \frac{1}{\sqrt{B_w B_h}} \sum_{x=0}^{B_w-1} \sum_{y=0}^{B_h-1} I_{ij}(x, y), \quad (8)$$

where DC_{ij} is the DC coefficient of the block located in the i -th row and the j -th column of the feature region, B_w and B_h are the width and the height of the block, and $I_{ij}(x, y)$ is the block in the i -th row and j -th column of the feature regions.

Step 5. Embed one-bit information of the watermark into DC_{ij} , according to Equation (9).

$$DC_{ij}^* = \begin{cases} (x - 0.5)\delta, & \text{if } (x + w_{ij}) \bmod 2 = 1, \\ (x + 0.5)\delta, & \text{if } (x + w_{ij}) \bmod 2 = 0, \end{cases} \quad (9)$$

where $x = \text{round}(DC_{ij}/\delta)$, $\text{round}()$ is the rounding function, DC_{ij}^* is the corresponding value after embedding the watermark into DC_{ij} , and δ is the quantization step to adjust imperceptibility and robustness.

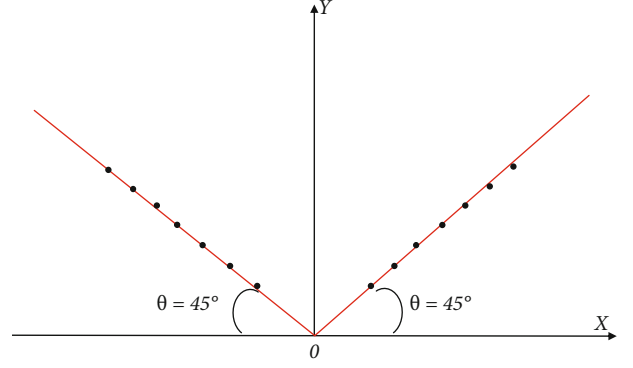


FIGURE 3: The template embedding.

Step 6. Obtain the watermarked block $I_{ij}^*(x, y)$ by the following equation [24]:

$$I_{ij}^*(x, y) = \frac{1}{\sqrt{B_w B_h}} \left(DC_{ij}^* - DC_{ij} \right) + I_{ij}(x, y) \quad (10)$$

When all the blocks in the current frame are embedded with the watermark, continue to do the same operation on the next frame.

4.2. Extraction Process. The template is extracted before the watermark extraction process, in order to recover the watermarked videos which may have been destroyed by geometric attacks, and then, the watermark is extracted from the recovered videos. In summary, the whole extraction process is shown in Figure 4.

4.2.1. Template Extraction. The whole procedures for extracting the template are shown below.

Step 1. Perform scene change detection on the watermarked video, according to Equation (2); obtain the Y components of the scene change frames.

Step 2. For each obtained Y component, pad it with zeros to a size of 3000×3000 and then apply the FFT to it.

Step 3. For the points on the two lines with an angle of 45 degrees between the radial and the coordinate axis of the DFT domain, all the local peak points (px_j, py_j) are extracted by the following formula:

$$\text{MagT}(px_j, py_j) - \text{LocalMean}(px_j, py_j) - \beta \text{Std}' \geq 0, \quad (11)$$

where j is the index of the peak points, $\text{MagT}(px_j, py_j)$ is the magnitude value of the point (px_j, py_j) , $\text{LocalMean}(px_j, py_j)$ is the average of the magnitude of 120 pixels adjacent to (px_j, py_j) , Std' is the standard deviation of the whole DFT spectrum of the selected frame, and β is the detection strength.

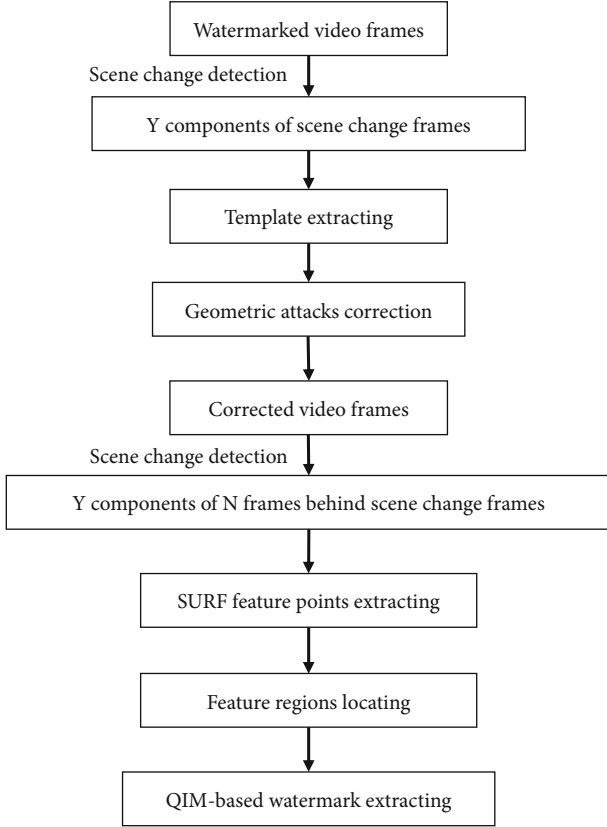


FIGURE 4: The extraction process.

Step 4. If there are at least 4 points that match the points in the original template line, it is considered that a matching line is found, and the matching rule is

$$px_j - Kx_i \leq \text{thresh}, \quad (12)$$

where px_j is the abscissa of the extracted peak point, x_i is the abscissa of the original template point, K is the scaling factor between 0.4 and 1.5, and thresh is the empirical value.

Step 5. Perform statistical analysis on the whole results to get the final matching result.

Step 6. According to the matching result of the template, the geometric attack correction is performed on the attacked video.

4.2.2. Watermark Extraction. The detailed steps of the watermark extraction are proposed below.

Step 1. Perform scene change detection on the watermarked video which is recovered by the template, based on (2), and then select the next N frames of each scene change frame for watermark extraction.

Step 2. Extract the SURF feature points on each selected frame, construct a $R \times R$ feature region with each point as the origin, and remove the points corresponding to the cross-border regions.

Step 3. Select the first ten points with the highest intensity, and the watermark is extracted from the feature regions corresponding to the selected points and their eight neighborhoods, according to the following equation:

$$w'_{ij} = \begin{cases} 1, & \text{if } \text{floor}\left(\frac{DC'_{ij}}{\delta}\right) \bmod 2 = 1, \\ 0, & \text{if } \text{floor}\left(\frac{DC'_{ij}}{\delta}\right) \bmod 2 = 0, \end{cases} \quad (13)$$

where DC'_{ij} is the DC coefficient of the image block in the i -th row and j -th column of the feature region, $\text{floor}()$ is the function of rounding down, w'_{ij} is the watermark information extracted from the image block, locating in the i -th row and j -th column of the feature region, and δ is the quantization step the same as that in the watermark embedding process.

Furthermore, the high decoding reliability of the QRCode is used to determine if the watermark is effectively extracted, which implies that when a QRCode can be successfully decoded by the decoder, the error rate of decoding is near zero. As a result, if the QRCode extracted from at least one feature region can be successfully decoded, the likelihood of treating it as the embedded watermark is close to 100 percent, and the ownership is established [11].

5. Experimental Evaluation

We evaluate the performance of the proposed video watermarking in this section. The experimental setup is introduced in Section 5.1. Section 5.2 verifies the effectiveness of the scene change-based frame allocation strategy. Section 5.3 compares the robustness between our algorithm and the state-of-the-art methods.

5.1. Experimental Setup. Test set. The test set includes 50 1080 P (1920×1080) and 50 720 P (1280×720) videos. They are all in the mp4 format, with a frame rate between 23.98 and 30 (frames per second) and a duration ranging from 90 to 360 (s).

Environments. The experiments were performed on a PC with 16 GB RAM and 3.4 GHz Intel Core i7 CPU, running on 64-bit Windows 10. The simulation software was Visual Studio 2010 with FFmpeg 2.1 and OpenCV 2.4.9.

Parameters. The number of frames behind every scene change frames to embed the watermark N is set to 10. The threshold of scene change Thre is set to 0.6. The middle frequency band $[f_1, f_2]$ is set to $[400, 478]$. The template embedding strength α is set to 30. The size of the QRCode to be embedded $W \times W$ is selected as 25×25 . The size of the feature region $R \times R$ is set to 200×200 . The quantization step δ is set to 60. The match threshold of the template thresh is set to 0.75. The detection strength β is set to 0.05.

Evaluation indexes. We evaluate the imperceptibility and the robustness of the algorithms by using the mean peak signal to noise ratio (MPSNR) and byte error rate (BER), respectively. The larger the MPSNR is, the better the imperceptibility is. On



FIGURE 5: The original frame and its corresponding geometric attacked frames.

TABLE 1: The imperceptibility comparison results between two different strategies.

Strategies	MPSNR (dB)
Proposed	47.94
TTWAS	41.64

the contrary, the smaller the BER is, the better the robustness is. Furthermore, they are calculated as Equations (14) and (15), separately.

$$\text{MPSNR} = \frac{1}{I} \sum_{i=1}^I 10 \log_{10} \frac{255^2 MN}{\sum_{m=1}^M \sum_{n=1}^N [F_i(m, n) - F_{iw}(m, n)]^2}, \quad (14)$$

where I is the number of the watermarked frames, M and N are the width and the height of the video, and F_i and F_{iw} are the i -th original frame and its corresponding watermarked frame.

$$\text{BER} = \frac{N(\text{errorBytes})}{N(\text{totalBytes})}, \quad (15)$$

where $N(\text{errorBytes})$ is the number of bytes inconsistent between the extracted watermark and the original watermark and $N(\text{totalBytes})$ is the number of bytes of the original watermark.

The types of geometric attacks. Padding, cropping, scaling, shielding, scaling and cropping, scaling and padding, and scaling and shielding are the mainly geometric attacks

TABLE 2: The comparisons of the robustness against geometric attacks between two different strategies.

Attack types	Parameters	Proposed (%)	TTWAS (%)
Padding	30%	0	-
	40%	0	-
	50%	0	-
Cropping	40%	0	-
	50%	0	-
	60%	0	-
Scaling	0.4	36	-
	0.5	2	-
	1.5	1	-
Shielding	25%	0	-
	36%	0	-
	49%	0	-
Scaling & cropping	0.8 & 5%	2	-
	0.8 & 10%	2	-
	0.8 & 20%	2	-
Scaling & padding	0.8 & 30%	1	-
	0.4 & 2%	79	-
	0.4 & 5%	79	-
Scaling & shielding	0.4 & 7%	88	-
	0.4 & 10%	89	-
	0.5 & 5%	1	-
Scaling & padding	0.5 & 10%	5	-
	0.4 & 5%	50	-
	0.4 & 10%	56	-

TABLE 3: The comparisons of the robustness against geometric attacks between two algorithms.

Attack types	Parameters	Proposed (%)	TFPA (%)
Padding	30%	0	0
	40%	0	0
	50%	0	0
Cropping	40%	0	0
	50%	0	0
	60%	0	0
	0.4	36	62
Scaling	0.5	2	4
	1.5	1	0
	25%	0	0
Shielding	36%	0	0
	49%	0	1
Scaling & cropping	0.8 & 5%	2	9
	0.8 & 10%	2	9
	0.8 & 20%	2	5
	0.8 & 30%	1	8
	0.4 & 2%	79	86
Scaling & padding	0.4 & 5%	79	86
	0.4 & 7%	88	88
	0.4 & 10%	89	93
	0.5 & 5%	1	7
Scaling & shielding	0.5 & 10%	5	8
	0.4 & 5%	50	74
	0.4 & 10%	56	76

used in this paper, and Figure 5 illustrates part of attacked frames of an original frame in [30].

5.2. Verifying the Effectiveness of the Scene Change-Based Frame Allocation Strategy. This subsection verifies the effectiveness of the proposed scene change-based frame allocation strategy by comparing it with the typical template and watermark allocation strategy (TTWAS) [22–25]. For a fair comparison, only the allocation manner of the template and the watermark of the two strategies differs, and the other settings are the same. We evaluate the effectiveness through the imperceptibility and the robustness.

The imperceptibility comparison results are shown as Table 1, and we can find that our strategy can enhance the imperceptibility efficiently. By using the TTWAS, the template and the watermark should be embedded in the same frame at the same time. However, in our proposed strategy, we allocate them to embed into different frames, so that the amount of the embedded data in a frame is reduced, and the imperceptibility becomes better.

Furthermore, the comparisons of the robustness against geometric attacks between the two different strategies are shown in Table 2, where “-” empresents that extracting the watermark fails. When the embedding strength of the two strategies is equal, which is set as before in Section 5.1, TTWAS cannot successfully withstand all of the attacks in

the table, while our approach has certain robustness. The mutual interference phenomenon that exists in TTWAS can be effectively eliminated through our strategy, for embedding the template and the watermark in different frames.

5.3. Comparisons of Robustness against Geometric Attacks. This subsection compares the ability of our algorithm and the baseline to recover the embedded watermarks in different geometric attacks. The typical and widely used location method of the feature point-based watermarking [17, 18] is selected as the baseline of the proposed video watermarking algorithm, which is named as TFPA (typical feature point-based algorithm).

The comparison results are shown in Table 3. In most cases, our algorithm outperforms the TFPA, due to introducing the template to assist the location of the feature points. After the video frames subjected to geometric attacks are restored, it is no longer necessary to use the scale information of the feature points when determining the feature regions, which reduces the locating error and improves the accuracy of the watermark extraction. However, our algorithm performs a bit worse against enlarging scaling, because the embedded templates may occasionally be undetected.

6. Conclusions

In this paper, a feature point-based video watermarking algorithm is presented, combining with the template. Because a video is made up of a series of images, it has more hosts for embedding the information. Based on this, a scene change-based frame allocation approach is proposed, which embeds the template and the watermark into different frames. This strategy can enhance the imperceptibility and avoid the mutual interference between the watermark and the template effectively. Furthermore, the insertion of the template reduces the location error of the feature points, improving the robustness against geometric attacks. In most cases, and the experimental results indicate that our algorithm outperforms the state-of-the-art approach. In future study, we will improve the robustness of our algorithm against more geometric attacks so that it can be applied to realistic scenarios.

Data Availability

The videos used to support the findings of this paper are subject to privacy or copyright, so that they cannot be shared.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Key R&D Program of China (2020YFB1406900) and the Key R&D Program of Shanxi (201903D421007). It was also the research achievement of the Key Laboratory of Digital Rights Services, which is one of the National Science and Standardization Key Labs for Press and Publication Industry.

References

- [1] M. Yan, S. Li, C. A. Chan, Y. Shen, and Y. Yu, "Mobility prediction using a weighted Markov model based on mobile user classification," *Sensors*, vol. 21, no. 5, p. 1740, 2021.
- [2] M. Yan, H. Yuan, Z. Li, Q. Lin, and J. Li, "Energy savings of wireless communication networks based on mobile user environmental prediction," *Journal of Environmental Protection and Ecology*, vol. 22, no. 1, pp. 206–217, 2021.
- [3] M. Asikuzzaman and M. R. Pickering, "An overview of digital video watermarking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 9, pp. 2131–2153, 2018.
- [4] X. Yu, C. Wang, and X. Zhou, "A survey on robust video watermarking algorithms for copyright protection," *Applied Sciences*, vol. 8, no. 10, pp. 1891–1917, 2018.
- [5] C. Li, Y. Yang, K. Liu, and L. Tian, "A semi-fragile video watermarking algorithm based on H.264/AVC," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8848553, 2020.
- [6] A. Cedillo-Hernandez, M. Cedillo-Hernandez, M. N. Miyatake, and H. P. Meana, "A spatiotemporal saliency-modulated JND profile applied to video watermarking," *Journal of Visual Communication and Image Representation*, vol. 52, pp. 106–117, 2018.
- [7] H. Zhao, Q. Dai, J. Ren, W. Wei, Y. Xiao, and C. Li, "Robust information hiding in low-resolution videos with quantization index modulation in DCT-CS domain," *Multimedia Tools and Applications*, vol. 77, no. 14, pp. 18827–18847, 2018.
- [8] S. Ponni alias Sathya and S. Ramakrishnan, "Non-redundant frame identification and keyframe selection in DWT-PCA domain for authentication of video," *IET Image Processing*, vol. 14, no. 2, pp. 366–375, 2020.
- [9] S. P. Alias Sathya and S. Ramakrishnan, "Fibonacci based key frame selection and scrambling for video watermarking in DWT-SVD domain," *Wireless Personal Communications*, vol. 102, no. 2, pp. 2011–2031, 2018.
- [10] X. Li, X. Wang, W. Yang, and X. Wang, "A robust video watermarking scheme to scalable recompression and transcoding," in *2016 6th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pp. 257–260, Beijing, China, 2016.
- [11] Z. Lv, Y. Huang, H. Guan, J. Liu, S. Zhang, and Y. Zheng, "Adaptive video watermarking against scaling attacks based on quantization index modulation," *Electronics*, vol. 10, no. 14, p. 1655, 2021.
- [12] K. Matsui and K. Tanaka, "Video-steganography: how to secretly embed a signature in a picture," *IMA Intellectual Property Project Procedure*, vol. 1, pp. 187–206, 1994.
- [13] P. Bas, J.-M. Chassery, and B. M. Macq, "Geometrically invariant watermarking using feature points," *IEEE Transactions on Image Processing*, vol. 11, no. 9, pp. 1014–1028, 2002.
- [14] C. -W. Tang and H. -M. Hang, "A feature-based robust digital image watermarking scheme," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 950–959, 2003.
- [15] X. Wang, P. Niu, H. Yang, C. Wang, and A. Wang, "A new robust color image watermarking using local quaternion exponent moments," *Information Sciences*, vol. 277, pp. 731–754, 2014.
- [16] H. Y. Lee, H. Kim, and H. K. Lee, "Robust image watermarking using local invariant features," *Optical Engineering*, vol. 45, no. 3, article 037002, 2006.
- [17] M. Kawamura and K. Uchida, "Sift feature-based watermarking method aimed at achieving IHC ver.5," in *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 381–389, Springer, Cham, 2018.
- [18] W. -L. Lyu, C. -C. Chang, T. -S. Nguyen, and C. -C. Lin, "Image watermarking scheme based on scale-invariant feature transform," *KSII Transactions on Internet and Information Systems*, vol. 8, no. 10, pp. 3591–3606, 2014.
- [19] X. Gao, C. Deng, X. Li, and D. Tao, "Local feature based geometric-resistant image information hiding," *Cognitive Computation*, vol. 2, no. 2, pp. 68–77, 2010.
- [20] B. Zhang, J. Wang, J. Wen, and Z. Tong, "A novel digital watermark against RST distortion based on surf," in *2010 IEEE International Conference on Information Theory and Information Security*, pp. 130–133, Beijing, China, 2010.
- [21] X. Liu, Y. Wang, J. Du, S. Liao, J. Lou, and B. Zou, "Robust hybrid image watermarking scheme based on KAZE features and IWT-SVD," *Multimedia Tools and Applications*, vol. 78, no. 5, pp. 6355–6384, 2019.
- [22] S. Pereira and T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE transactions on image Processing*, vol. 9, no. 6, pp. 1123–1129, 2000.
- [23] S. Pereira and T. Pun, "Fast robust template matching for affine resistant image watermarks," in *International Workshop on Information Hiding*, pp. 199–210, Springer, Berlin, Heidelberg, 2000.
- [24] X. Qi and J. Qi, "Improved affine resistant watermarking by using robust templates," *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 405–408, 2004.
- [25] T. Tokar and D. Levicky, "Robust watermarking of gray scale images by using synchronization templates," in *2007 17th International Conference Radioelektronika*, Brno, Czech Republic, 2007.
- [26] Q. Su and B. Chen, "Robust color image watermarking technique in the spatial domain," *Soft Computing*, vol. 22, no. 1, pp. 91–106, 2018.
- [27] C. Wang, X. Li, M. Xu, J. Wang, and W. Wan, "Blind photograph watermarking with robust defocus-based JND model," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8892349, 2020.
- [28] Y. Huang, B. Niu, H. Guan, and S. Zhang, "Enhancing image watermarking with adaptive embedding parameter and psnr guarantee," *IEEE Transactions on Multimedia*, vol. 21, no. 10, pp. 2447–2460, 2019.
- [29] H. Sadreazami, M. O. Ahmad, and M. Swamy, "Multiplicative watermark decoder in contourlet domain using the normal inverse Gaussian distribution," *IEEE Transactions on Multimedia*, vol. 18, no. 2, pp. 196–207, 2016.
- [30] <https://media.xiph.org/video/derf/>.