An Improved Histogram Based Boosting Detection Rate Video Watermarking Algorithm

GuangXi Chen
Guilin University of Electronic Technology School of Computer Science and Information Security
Guilin, China
chgx@guet.edu.cn

TianRen Cai
Guilin University of Electronic Technology School of Computer Science and Information Security
Guilin, China
Tsinghua University Department of Computer Science and Technology
Beijing, China
thomasctr001@163.com

DaoShun Wang
Tsinghua University Department of Computer Science and Technology
Beijing, China
daoshun@mail.tsinghua.edu.cn

XiangHui Zhao
China Information Technology Security Evaluation Center
Beijing, China
zhaoxianghui@mail.itsec.gov.cn

Yong Huang
Guangzhou University School of Computer Science and Education Software
Guangzhou, China
79031404@qq.com

YiJun Zhan
Guilin University of Electronic Technology School of Computer Science and Information Security
Guilin, China
1440656670@qq.com

Abstract**—**The existing histogram based video watermarking algorithm with temporal modulated is robust to combined attacks, but the watermark detection rate is not high due to watermark cannot embedded to the smoothness and still areas effectively. To increase the watermark detection rate, in this paper, we proposed the improved algorithm of shot segmentation first and then propose an improved video watermarking algorithm which firstly construct the watermark template in each frame video in the same shot through computing block based histogram and selecting the position of the relative high variance. Then we embed the watermark template into the video frame by temporal modulation without changing the destination of the shot group of the consecutive frames. The watermark sequence is extracted by comparing the correlation distribution of video frame and corresponding watermark template in the time domain. Experimental results demonstrate that the proposed algorithm is robust to recording attacks and guarantee the watermarking video quality at the same time, besides the watermark sequences can embedded to the smoothness and still areas effectively, and the watermark detection rate can increase by about 10% than previous methods.

Keywords*-video watermarking; histogram; shot segment; HVS; watermark detection rate*

*****

I. INTRODUCTION

Histogram is widely used in video watermarking algorithm because it calculates the number of pixel points and ignores the location of pixel points. In order to solve the problem of geometric attack and reversible video watermarking, the existing histogram video watermarking algorithm can be divided into two categories: the Gray-level histogram modulation in the space domain and the frequency coefficient's histogram modulation in the transform domain. In the video watermarking based on the grayscale histogram modulation in the space domain, the histogram is mainly calculated by the method of motion estimation and prediction error. Zeng et al. [1] calculated prediction error histogram of consecutive frames by motion estimation of neighboring frames, and then the peak point of the histogram is expand to embed the reversible watermark, which can improve the watermark capacity and the watermark video quality. The following year, in [2], they changed the way of modulation histogram, which expand the position of the prediction error to expand the error, so as to improve the watermark capacity and better visual quality more. Vural et al. [3] used a recursive histogram modification to calculate motion compensated frame interpolation errors, and embedding watermark into odd and even frames crossly, the scheme has good performance in watermark capacity and robust to video distortion attacks. Zheng et al. [4][5] and Hu et al. [6] embedded watermark sequences by changing the relationship of bins of image histogram, which has good robustness to the geometric attacks.

The video watermarking based on the frequency coefficient's histogram modulated in the transform domain can be divided into: wavelet transform, integer wavelet transform, DCT Transform, DFT transform and so on. Alavianmehr et al. [7] calculated sub band image of the luminance component and divided it to non-overlapping blocks, and then applies histogram to calculate different value of consecutive block. The watermark is embedded into the blocks based on a multi-level shifting mechanism of the histogram. The scheme has great watermark capacity and robust to H.264 /AVC compression attacks. Chen et al. [8] constructed the histogram of the average DC coefficients of the video frames in the time domain, and then embeds the watermark sequences by changing the relationship between consecutive groups of the histogram, which has strong robustness to the geometric attack, frame loss and frame average. Gao et al. [9] applied integer wavelet transform to divide video frames into low-frequency and middle-frequency domains and uses different algorithms to embed watermarks. Based on the motion information of the video, the watermark is embedded in the middle and high frequency domain and changed the consecutive coefficients of the low frequency sub band histogram to embed watermarking in the low frequency domain, and the scheme has strong robustness to the geometric attack, too.

The above histogram based video watermarking algorithms are not designed to be robust to recording attacks. Because embedding the watermark in the transform domain is rather time consuming, this paper does not consider using those kinds of methods to embed video watermarking. However, among the space domain methods, Do et al. [10] proposed a video watermarking method based on histogram modulation. Their method first generates a watermark template from the histogram of a single shot video frame, and then uses it to embed watermarking in video frames by way of temporal modulation. This method has strong robustness to geometric and camcorder recording attacks, but there is a drawback: the selection of texture detection and motion detection makes it unable for watermarks to be embedded effectively in smooth and still regions.

To address the drawback from Do et al.'s method [10], Gaj et al. [11] applied the method of scale invariant feature transform (SIFT) to segment the single shot before embedding and extracting watermark sequences. The three-dimensional and two level wavelet transform is used to adjust the DC coefficients of the SIFT feature points in the left and right parts of the LLL sub-band to embed the watermark sequences. The watermark sequences were extracted by comparing the energy value of SIFT feature points from two parts of the video frames. However, since this method does not use histograms, the experimental results of this paper are only compared with Do et al.'s method.

In this paper, we propose an improved shot segmentation scheme and a new video watermarking algorithm based on histogram modulation which is not computationally intensive. Experimental results show that the watermark video has better visual perception, can be effectively embedded in a rough and still regions, is robust to camcorder recording attacks, and has a higher detection rate.

The rest of the study is organized as follows. The temporal modulation based watermark pattern and Do et al.'s algorithm is reviewed in Section 2. The improved shot segmentation algorithm and the proposed algorithm is discussed in Section 3, and its performance test discussed in Section 4. Finally, Section 5 summarizes the study's contribution.

II. Theoretical Background and Research Motivation

A. *Temporal Modulation based video watermarking algorithm pattern*

Among video watermarking algorithms which are robust to recording attacks, Leest et al. [12] present a method based on temporal modulation for embedding watermarks, as shown in Fig. 1. Experimental results show that the video watermarking algorithm can resist camcorder recording attacks.

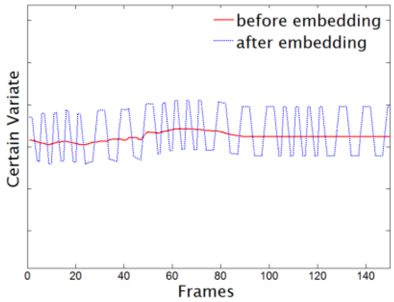


Figure 1. The real line is the video sequence before the watermark embedding; the dotted line is the video sequence after the watermark embedding

B. *Do et al.'s method Analysis Based on Histogram Temporal Modulations*

Based on the above video watermarking algorithm model, many researchers also proposed their own ideas. Among them, Do et al. [10] applied histograms to change the correlation of original video frames to embed watermark sequences in the time domain. Specifically, they applied a histogram to generate a frame's watermark template whose embedding power depends on the texture detection and motion detection proposed by Leest et al.'s method [12]. Then the temporal modulation is used to embed the watermark according to (1) rule.

$$F_w(\underline{n},k)=\begin{cases}F(\underline{n},k)-S[m]P(\underline{n},k)W(\underline{n},k), & k=5(m-1)+1,2\\ F(\underline{n},k), & k=5(m-1)+3\\ F(\underline{n},k)+S[m]P(\underline{n},k)W(\underline{n},k), & k=5(m-1)+4,5\end{cases} \quad (1)$$

Where $F(\underline{n},k)$ and $F_w(\underline{n},k)$ show the k-th frame before the watermark embedding and after the watermark embedding respectively. $\underline{n}$ is the two-dimensional vector and represents the pixel position on the image. $S[m]$ represents watermarking bit -1 or 1. m represents the m-th watermarking bit. $P(\underline{n},k)$ represents watermarking embedding power. $W(\underline{n},k)$ represents watermarking template.

1) *Embedding and Extracting of Video Watermark*

Firstly, the histogram of each video frame is computed and divided into different areas. The watermark template is generated according to the position of the different areas,

The watermark template and the embedding power collectively is called a "power watermark template". The template is then used to change the correlation distribution between each frame and the power watermark template to embed the watermark. A brief review of the Do et al.'s algorithm is as follows.

In the watermark embedding process, a crude human visual system that depends on the texture detector and the motion detector is applied to adjust the embedding power. The texture detector is calculated as (3) by using the R-, G-, and B-component of each frame according to the two dimensional convolution of the given Laplace High-pass filter L (2) respectively.

$$\square\ \square L=\frac{1}{9}\begin{bmatrix}-1 & -1 & -1\\ -1 & 8 & -1\\ -1 & -1 & -1\end{bmatrix} \quad (2)$$

$$H_T(\underline{n},k)=|(F_R(\square,k)*L)(\underline{n})|+|(F_G(\square,k)*L)(\underline{n})|+|(F_B(\square,k)*L)(\underline{n})| \quad (3)$$

Where $F_R(\underline{n},k)$, $F_G(\underline{n},k)$, and $F_B(\underline{n},k)$ represent R-, G-, and B- component of the k-th video frame. H_T represents texture detector and * represents a two dimensional convolution operation.

The motion detector is calculated by the difference of the luminance component of two consecutive frames. It is as follows:

$$H_M(\underline{n},k)=\left|F_Y(\underline{n},k)-F_Y(\underline{n},k-1)\right| \quad (4)$$

Where $F_Y(\underline{n},k)$ represents Y- component of the k-th video frame. H_M represents the motion detector.

The embedding power is calculated by adding a texture detector and a motion detector. Which is as follows:

$$P(\underline{n},k)=\alpha\cdot\log_{10}\left(H_T(\underline{n},k)+H_M(\underline{n},k)+1\right) \quad (5)$$

Where α represents adjustable constants, $P(\underline{n},k)$ represents the embedding power.

The watermark embedding algorithm steps are as follows:

Input: Original single shot video and watermark sequences $w\in\{0,1\}$

Output: Embedded single shot video

Embedding Process:

1. Firstly, convert the video to YUV format and process it to the 1st consecutive five frames of the video frame.
2. Then, apply a histogram to compute the consecutive five frames of the video frame. Through the histogram, they divide the video frame into two areas A and B.
3. For a video frame with N pixels, the number of pixels in area A and area B is 0.475N respectively. The pixel points in the middle of the histogram can improve the accuracy of the watermark detection.
4. Construct a watermark template W: with video frame pixel values corresponding to histogram of area A with position set to 1, and area B with position set to -1. Area C is set to 0. This matrix then becomes the watermark template.
5. Applying the texture detector and motion detector method, calculate the embedding power P, and record the final power watermark template as $W(\underline{n},k)\cdot P(\underline{n},k)$.
6. Embed watermarks according to (1) rules.
7. For the 2nd consecutive five frames, the 3rd consecutive five frames … until the n-th consecutive five frames, repeat 2-5 steps to embed watermark sequences until the end of the video. Finally, we get the embedded single shot video.

Before extracting the watermark, Do et al.'s method proposes the method of image areas detection, which reduces the influence of the part outside the image during the video recording. The watermark extracting algorithm steps are as follows:

Input: Embedded single shot video

Output: Extracted watermark sequences $w\in\{0,1\}$

Extracting Process:

1. For the single shot video after embedding the watermark, the watermark is extracted from the 1st video frame with a consecutive five frames defined as a watermarking group of pictures.
2. For the 1st consecutive five frames, apply the embedding algorithm to calculate the 2nd frame and 4th frame of the power watermark template $W(\underline{n},2)\cdot P(\underline{n},2)$ and $W(\underline{n},4)\cdot P(\underline{n},4)$. Then, respectively, calculate the image and its watermark template correlation coefficient called $Corre(\underline{n},2)$, and $Corre(\underline{n},4)$.
3. Extract the watermark according to the following rules:

$$Corre(\underline{n},4)-Corre(\underline{n},2)\begin{cases}>0, & w=1\\ <0, & w=0\end{cases} \quad (6)$$

4. Repeat steps 2 and 3 until the end of the video for the 2nd consecutive five frames, the 3rd consecutive five frames … and the n-th consecutive five frames, until finally all watermark sequences W are extracted.

By embedding and extracting the algorithm, the result is better for the video shot with rich color, more texture and changeable frames. By calculating the average PSNR value of the video frame before and after embedding the watermark, we evaluate visual perception of embedding the video. The calculation formula of PSNR value is shown in (7).

$$\text{PSNR}=10\times\lg\left(\frac{M\times N\times 255^2}{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\left(F_w(I_{i,j})-F(I_{i,j})\right)^2}\right) \quad (7)$$

Where M and N is size of video frame. $F(I_{i,j})$ and $F_w(I_{i,j})$ is the value of the luminance component before and after embedding in the position i and j of the video frame respectively. That uses a standard single shot video to calculate the average peak signal-to-noise ratio of the embedded video:

$$PSNR=45.13dB$$

In conclusion, the embedded video in this watermarking algorithm has excellent quality, the following is an analysis.

2) Do et al.'s method [10] analysis

In the video watermarking method, Do et al.'s method is robust to geometric attacks, compression attacks and camcorder recording attacks. However, Experimental results show that the method has a drawback: Because embedding power of the algorithm depends on the texture detector and motion detector, the watermark cannot be embedded in the smooth and still. This leads to a decrease in the watermark detection rate.

For the drawback, on the one hand, if the shot is a still image, the pixel value of the 5-frame image are similar, so that the motion detector in the embedding power is 0. This failure of the motion detector leads to significant decrease in the embedding power. On the other hand, R-, G-, B- components of the image are convolution operations with embedding power calculated by a Laplacian high-pass filter. Therefore, if the image is still and smooth, the embedding power calculated through convolution will also be very low. A more intuitive

International Journal on Future Revolution in Computer Science & Communication Engineering
Volume: 4 Issue: 4
ISSN: 2454-4248
757 – 766

explanation is given in the following example which applies the texture detector to calculate an 8×8 image block:

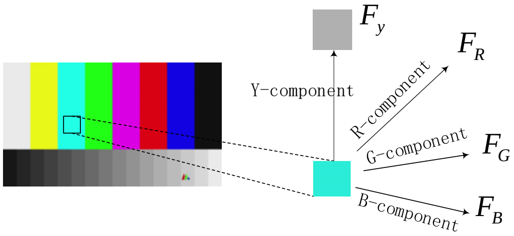


Figure 2. The Y- component value of the image with smooth regions and R-, G-, B- component

The values of the Y-, R-, G-, and B- component are:

$$F_y = \begin{bmatrix} 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \\ 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \\ 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \\ 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \\ 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \\ 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \\ 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \\ 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \end{bmatrix}$$

$$F_R = \begin{bmatrix} 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 \\ 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 \\ 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 \\ 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 \\ 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 \\ 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 \\ 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 & 485.8420 \\ 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 & 487.2440 \end{bmatrix}$$

$$F_G = \begin{bmatrix} -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 \\ -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 \\ -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 \\ -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 \\ -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 \\ -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 \\ -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 & -17.6155 \\ -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 & -18.3296 \end{bmatrix}$$

$$F_B = \begin{bmatrix} 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 \\ 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 \\ 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 \\ 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 \\ 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 \\ 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 \\ 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 \\ 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 & 360.2880 \end{bmatrix}$$

Where the value of the R-, G- and B- components is between 0 and 255, but the value of the R-, B- component is greater than 255, and G- component is less than 0, the R-, B- components become 255 and the G- component 0. Then, the algorithm can get the last R-, G-, B- component matrix. According to the equation (4): texture detectors $H_T(\underline{n}, k) \approx 0$, we can see that for the smooth images, texture detectors are almost ineffective.

Therefore, for the single shot video with the smooth and still regions, the embedding power is too small, and is too difficult to embed the watermark into the video.

III. The Proposed Algorithm

Watermarks embedded into a single shot video using the temporal modulation method improve the detection rate of watermark embedding. Therefore, we first discuss the improved shot segmentation algorithm.

A. Improved Shot Segmentation Algorithm Based on Histogram Correlation

In this paper, the improved shot segmentation algorithm based on histogram correlation is proposed. It can improve the detection rate and the accuracy rate of watermarks when extracting the watermark. Firstly, it calculates the histogram to ensure that the pixels that have a value greater than 50 have a specific proportion of the whole frame pixel count, so that the watermark has enough area to be embedded. Then, it calculates the correlation of the histogram with the video frame. If the correlation of the histogram of two consecutive frames is greater than a certain threshold value, these video frames belong to the same shot.

The histogram can be considered as the image pixel level function which then describes the number of occurrences of this pixel in the image. The horizontal axis indicates the number of pixel values and ordinate axes indicates the number of occurrences of this pixel in the image. (In this paper, we use a grayscale histogram, whose name is derived from using pixel values from the gray value regions of an image.)

The structure of the grayscale histogram of an image is as follows:

$$H(I) = \{h(x_1), h(x_2), ..., h(x_N)\} \tag{8}$$

Where n represents that the histogram has N gray values and $x_1, x_2, ..., x_N \in [0, N]$. In this paper, N takes 255 and $h(x)$ represents the times of occurrences of each gray value.

The histogram correlation of the image I_1 and I_2 the histogram is as follows:

$$d\left(H(I_1), H(I_2)\right) = \frac{\sum_N \left(H(I_1) - \overline{H(I_1)}\right)\left(H(I_2) - \overline{H(I_2)}\right)}{\sqrt{\sum_N \left(H(I_1) - \overline{H(I_1)}\right)^2 \left(H(I_2) - \overline{H(I_2)}\right)^2}} \tag{9}$$

Where $\overline{H(I_k)} = \frac{1}{N}\sum_{i=1}^{N} h(x_i)$

Improved shot segmentation algorithm based on histogram correlation is as follows:

Input: Original with multiple shot video V_0

Output: Single shot videos $shot_1, shot_2 \cdots\cdots shot_n$

Algorithm Process:

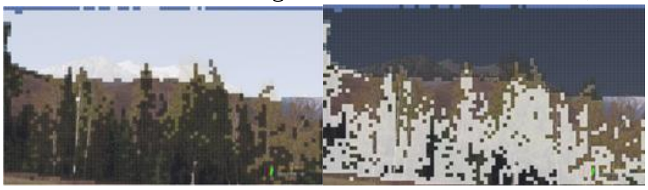
1. Starting from the first frame I_1 of the original video V_0, the algorithm calculates the histogram $H(I_1)$ of the video frame.
2. It then calculates the total number N_0 of pixels in this video frame and the number N_x of pixel with gray value in the range of $[50, 255]$. If their ratio is greater than the threshold, that is: $N_x / N_0 > \theta$, this frame is the first frame of a single shot video frame.

International Journal on Future Revolution in Computer Science & Communication Engineering **ISSN: 2454-4248**
Volume: 4 Issue: 4 **757 – 766**

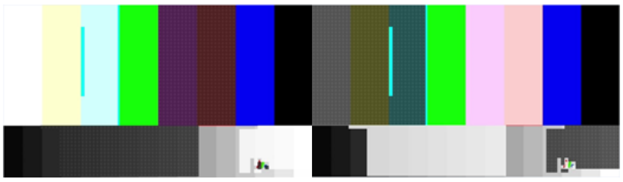
3. From the 2nd frame (i.e. k=2) until the end of the video, the correlation of the consecutive frames I_{k-1}, I_k with the histogram $H(I_{k-1})$, $H(I_k)$ is calculated. If the correlation is greater than the threshold, that is: $d(H(I_{k-1}),H(I_k)) > \varphi$, the video frames I_{k-1} and I_k, belong to the same shot ($shot_1$).
4. If the correlation is less than the threshold, the video frames I_{k-1} and I_k do not belong to the same shot. Steps 2 and 3 are repeated for the video frame I_k to determine each shot of the video ($shot_1, shot_2 \cdots\cdots shot_n$).

B. Selection of Embedding Areas

In this paper, the whole frame is divided into pixel blocks. We then calculate the histogram of the block mean. Here,



(a)



(b)

Figure 3. (a) and (b) is different shot's 2nd frame and 4th Frame embedding areas after block and variance calculation

taking the 8×8 pixel block as an example, the block method is shown in Fig. 4.

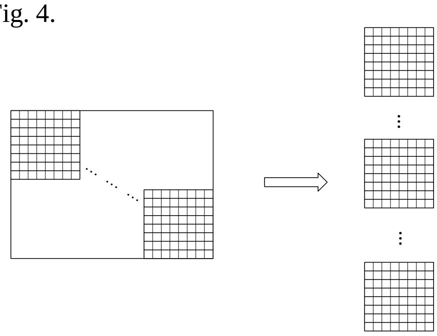


Figure 4. The whole frame is divided into 8×8 pixel block

In addition, the visual perception of the human eye is a masking local effect, which is influenced by background brightness, texture complexity and signal frequency. For the image, the main characteristics of the human vision system are as follows: Luminance characteristic, frequency domain characteristic and image type characteristic. In this paper, the brightness characteristics and image type characteristics are considered. By selecting the histogram embedding areas with relatively larger variance, the perception of video changes is effectively reduced. The improved embedding areas after block and variance calculations is shown in Fig. 3.

Fig. 3. (a) (b) of the black and white regions correspond to the "-1" and "+1" areas of the watermark template respectively; besides these areas, 60% of the whole frame is accounted for. It can be seen that the embedding area is less than Do et al.'s method [10], and, embedding the watermark into video frames using the block method, improves the effect of visual masking, especially in high-definition video. It can reduce the video flicker caused by changing the pixels in the whole frame. Fig. 3. (b) shows good performance in videos containing smooth and still regions.

In addition, you can see that the pure black regions of Fig. 3 (b) don't change. Because, in this paper, we chose the pixel blocks with gray values greater than 50 to embed the watermark. According to Watson's human visual system, the human eye is sensitive to brighter areas. Experimental results show that recorded video in dark environments, with a background area (if any) of the recorded video less than 50, can improve the visual effect further; thus, improving the watermark extraction rate after the camcorder recording attack.

However, if the whole frame image is darker and the gray value is mostly less than 50, the embedding area will be very small and will consequently make the watermark embedding area very small. This would affect watermark extraction rate, which is why we proposed the advanced shot segmentation algorithm to ensure the video frame quality of the shot. The proportion of the embedding video frame that has pixel gray values greater than 50 can then be higher than a specific threshold.

C. Embedding Process

The video frame blocks are processed and the block histogram is calculated according to the above embedding areas selection method. The embedding region is then guaranteed to be 60% of the whole frame by calculation (see the embedding process). (Experimental results show that the ratio can guarantee the visual masking and higher detection rate). Thus, a new watermark template is generated.

For the specific embedding method: a watermark is embedded in consecutive frames in the same shot using the temporal modulation method. If the watermark bit is "0", the front $(k-1)/2$ frames (1 ~ $(k-1)/2$ frame) plus the watermark template and the back $(k-1)/2$ frame ($(k-1)/2$ ~ k frame) minus the watermark template; If the watermark bit is "1", the front $(k-1)/2$ frame (1 ~ $(k-1)/2$ frame) minus the watermark template W_p, and the back $(k-1)/2$ frame ($(k-1)/2$ ~ k frame) plus the watermark template W_p. The embedded mathematical equation is shown as follows (where the watermark bit is "0", $S[m]$ takes -1; the watermark bit is "1", $S[m]$ takes 1):

$$F_w(I) = \begin{cases} F(I_m) - S[m]W_p, & m = 1,2,\ldots,(k-1)/2, \\ F(I_m), & m = (k+1)/2, \\ F(I_m) + S[m]W_p, & m = (k+3)/2,\ldots k. \end{cases} \quad (10)$$

Finally, in order to reduce the influence of the noise introduced in the recorded video, the Gaussian low-pass filter is used to filter the video before embedding and extracting, which can improve the robustness of the algorithm. The watermark embedding algorithm steps are as follows:

Input: Original video V_0 with multiple shots and watermark sequences $w \in \{0,1\}$

Output: Embedded video with multiple shots V_w

Embedding Process:

1. Apply the improved shot segmentation algorithm to divide the YUV format video into t single shot video V_0 , and embed the watermark into the Y-component.
2. Firstly, a single shot (V_t) of continuous k frames passes through a Gaussian filter, and then each frame of $N \times N$ block is calculated based on a mean histogram $H(I_1), H(I_2), ..., H(I_k)$. The areas ($[50, 255]$) are called the histogram embedded areas, defined as F . Where the proportion of embedded areas accounted for, ω , is the total number of blocks.
3. Watermark Template selection: First, F is divided into two parts which account for 0.4 of the completely embedded areas. Additionally, the middle of the remaining area, which accounts for 0.2 of the completely embedded areas, is a protection area. For A and B areas, the variance of pixel block is calculated respectively, and the block with relatively larger variance as embedded areas η , and set the value 1 and -1 respectively is chosen, while the other position is 0. $\omega \cdot 0.4 \cdot \eta \cdot 2 = 0.6 \Rightarrow \omega \cdot \eta = 0.75$ Where 0.6 of the equation indicates that the final embedding areas accounts for 60% of the whole frame.
4. Embedding watermarks to consecutive k frames of a single shot in block $N \times N$ is the base method according to (11) rules. This process is used until the end of the shot.
5. Embedding the watermark is done repeating Steps 2-4 for all shots (i.e. $t \in [1, n]$), resulting in embedded video with multiple shots V_w .

D. Extracting Process

Based on the conclusion of Do et al.'s algorithm, the watermark can be extracted by comparing the correlation between the 2nd frame and the corresponding watermark template of the embedding video and the correlation between the 4th frame and the corresponding watermark template of the embedding video. For the consecutive k frame, it is the comparison between $(k-1)/2$ th frame and $(k+3)/2$ th frame. However, in order to enhance time synchronization, this paper compares correlation between $(k-1)/2$ th frame and $(k+1)/2$ th frame and compares correlation between $(k+1)/2$ th frame and $(k+3)/2$ th frame. Besides comparing correlations, we need to remove these pixels from the frame. During the watermark embedding process, we embed the pixels in $[50, 255]$. It excludes other pixels which can change the histogram distribution of the single frame. The watermark extracting algorithm steps are as follows:

Input: Embedded video with multiple shots V_w

Output: Extracted watermark sequences $w \in \{0,1\}$

Extracting Process:

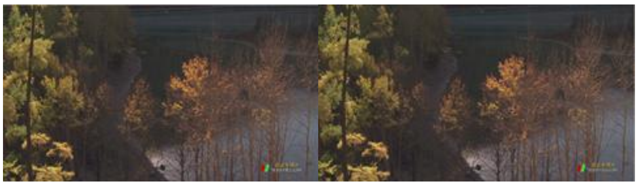
1. For the YUV format of embedded watermark video's Y- component, we first apply the improved shot segmentation algorithm to divide the video into each shot, then extract the watermark from each shot respectively.
2. Consecutive k frames of the single shot pass through a Gaussian filter, and then calculate the continuous k frames $(k-1)/2$ th frame $F_{(k-1)/2}$ and $(k+1)/2$ th frame $F_{(k+1)/2}$ and $(k+3)/2$ th frame $F_{(k+3)/2}$ of the power watermark template $W_{(k-1)/2}$, $W_{(k+1)/2}$ and $W_{(k+3)/2}$. The calculation method is the same as the embedding algorithm.
3. Calculate the correlation coefficient $C_{(k-1)/2}$ between $F_{(k-1)/2}$ and $W_{(k-1)/2}$, the correlation coefficient $C_{(k+1)/2}$ between $F_{(k+1)/2}$ and $W_{(k+1)/2}$, the correlation coefficient $C_{(k+3)/2}$ between $F_{(k+3)/2}$ and $W_{(k+3)/2}$. Then calculate the value of $C_{(k+1)/2} - C_{(k-1)/2}$ and $C_{(k+3)/2} - C_{(k+1)/2}$, if these value are both greater than 0, then the watermark bit is "1". If these value are both less than 0, then the watermark bit is "0".
4. Repeat the Steps 2 and 3 for the next consecutive k frames of the shot until end of the shot, and the watermark of the shot will be extracted.
5. Repeat Steps 2, 3, 4 for each shot until the watermark sequences of each shot is extracted.

In order to ensure the synchronization of time domain during extracting process, we apply the method of synchronization sequence which embeds the synchronization sequence before the watermark information sequence. When two synchronization sequences are detected, the watermark information sequence between two synchronous sequences are detected. If the number of detected watermark information sequences is exactly the same number of real watermark information sequences, it is considered that the watermark information sequences is embedded. It then applies the above extracted algorithm to extract the watermark information sequences. In addition, in order to correct the watermark sequences of 1 to 2 bit errors, we apply error correction code to synchronize sequences and improve detection accuracy of the watermark.

IV. Experiment And Result Analysis

This is a discussion about the size of pixel blocks. If the block is selected as a 2×2 block or a 4×4 block, the variance chance of the before and after recording is large due to the small size of block and the effect of extracting the watermark after poor recording. If the block is selected as 16×16 block or larger, the requirement of space synchronization is higher.

(a)



(b)

(c)

(d)

(e)

Figure 5. Screenshots of five shots video before embedding (left) and after embedding (right) watermark

Once the space synchronization is lost, the detection accuracy of the watermark is greatly reduced. Therefore, we chose the 8 ×8 block size.

In addition, in Second 1, we mentioned the latest advanced method [11] to improve Do et al. [10] method's performance, but this method is not based on histograms, we only compare it to Do et al.'s method [10].

In this experiment, our tested videos are 5 high-definition H.264/AVC videos with single shots of size 1920 by 1080 pixels, length 12 seconds and 500 frames, rate 25 fps, and bit rate 10Mbps and 1 video with multiple shots, length 13 minutes and about 20000 frames, which other parameters are similar as the above 5 single shot videos. We set k=5, which means consecutive 5 frames is a watermark bit. The total watermark sequences is 20-bit, including 10-bit synchronization sequences and 10-bit watermark information sequences, so each single shot short video can embed 5 watermark sequences (100 bits) and theoretically, 1 multi-shots long video can embed 200 watermark sequences (4000 bits). Actually, when we detected 2 synchronization sequences and the number of detected watermark information bit is exactly the same number of the real watermark information bit, we started to extract the watermark information bit. Therefore, 4 watermark information sequences can be extracted from single shot video. In the multi-shot video, due to some frames with change of shots are skipped by the advanced shot segment algorithm, the actual extractable watermark information bit depends on the proportion ω of embedded areas in the whole frame. This experiment set ω=0.8 , and 112 watermark information sequences can be extracted from the video with multiple shots in the experiment.

In this experiment, we evaluate the comprehensive performance of the algorithm in the following three aspects: invisibility, robustness and time complexity. The robustness has two indexes: the accuracy rate is used to measure the correct rate of the watermark sequences after the attack, and the calculation equation is (11); the detection rate is used to measure the extracted rate of watermark sequences after the attack, and the calculation equation is (12).

$$accuracy\ rate=\frac{correct\ watermark\ bit}{whole\ detected\ watermark\ bit} \tag{11}$$

$$detection\ rate=\frac{correct\ watermark\ bit}{whole\ correct\ watermark\ bit} \tag{12}$$

A. Analysis of Invisibility

From the subjective, watermark video quality is basically unaffected and almost imperceptible. As shown in Fig. 5, the following is a screenshot of the test video before and after embedding watermark:

Objectively, this paper applied average PSNR value of the video frame to evaluate the visual quality of the video after embedding the watermark and compared the average PSNR between the watermark video of the proposed algorithm and paper [10] algorithm as shown in Table 1:

TABLE I. AVERAGE PSNR

Video Sequence	Average PSNR	
	Proposed（HD）	***Paper [10] (SD)***
Shot.1	43.67dB	-
Shot.2	40.98dB	42.68dB
Shot.3	43.18dB	44.10dB
Shot.4	46.14dB	42.87dB
Shot.5	40.81dB	39.69dB

The method of the paper [10] cannot embed watermark to the shot.1 video effectively, so the PSNR value cannot be computed. Although average PSNR of the shot.2 or shot.3 video embedded by the proposed algorithm is lower than the PSNR of that video embedded by the paper [10] algorithm, the PSNR value of all videos embedded by the proposed algorithm is greater than 28, which can be considered invisible to naked eyes.

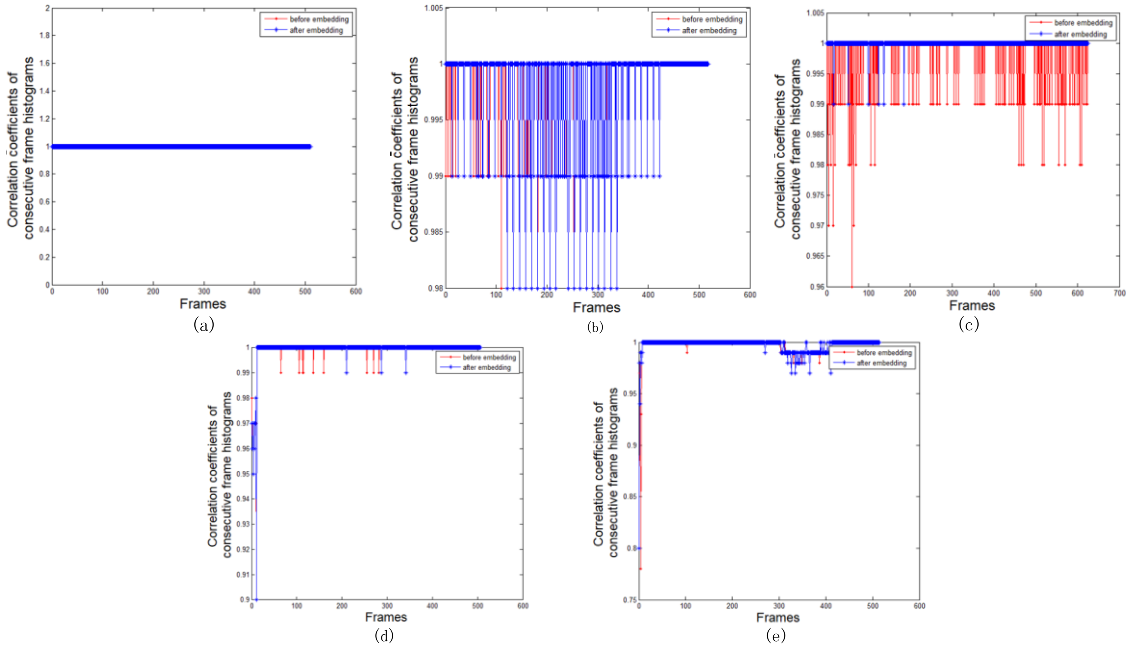


Figure 6. Five shots consecutive frame histogram's correlation

B. *Analysis of Robustness*

Firstly, the geometric attacks robustness of the watermark template based on histogram and 8 × 8 block variance calculation is discussed without embedding the watermark. That selects five video frames, which are the five frames shown above, to compare the difference of correlation between the original image and the watermark template of this image and difference of correlation the between the images under the geometric attacks and the watermark template of the image under the attack. It is analyzed in the following table:

TABLE II. THE CORRELATION DIFFERENCE BETWEEN IMAGE AND ITS WATERMARK TEMPLATE UNDER GEOMETRIC ATTACKS

Geometric distortion	**Correlation diff**				
	Seq1	***Seq2***	***Seq3***	***Seq4***	***Seq5***
Scaling Down(1024 ×768)	0.009	0.014	0.022	0.002	0.025
Scaling Down(800× 600)	0.006	0.004	0.022	0.008	0.012
Cropping 5%	0.003	0.005	0.010	0.004	0.003
Cropping 10%	0.007	0.010	0.012	0.007	0.006
Rotate 3°	0.043	0.023	0.054	0.041	0.021
Rotate 5°	0.065	0.046	0.067	0.054	0.045
Addition 5%	0	0	0	0	0
Addition 10%	0	0	0	0	0
Addition 15%	0	0	0	0	0

As you can see from Table 2, shrinking the size, cropping the attack, and increasing the black area do not affect the relevance of the image to the watermark template. However, after the rotation attack, the correlation difference value is large.
This is because this article has taken the strategy of the statistical block histogram and this strategy can relieve the flicker phenomenon. But it causes the spatial synchronization to descend and increase the difference value of correlation before and after the rotation attack. However, the result of this experiment is only a reference factor. The video watermarking algorithm can resist the attack depending on whether the watermark can be retained after the attack. So the robustness of the watermarking algorithm is then tested.

Secondly, the effectiveness of the improved shot algorithm based on histogram correlation was tested. In order to verify whether the consecutive frames of the single shot video are still within the threshold range, we applied the line chart to compare the correlation of the consecutive frames of five single shot video before and after embedding the watermark. In particular, the consecutive frame correlation threshold of the same shot in this experiment is set to be 0.7, that is to say, the correlation coefficient of the consecutive frames of the five single shot video is greater than 0.7. It is now necessary to verify whether the correlation coefficient of the consecutive frames of the embedded single shot video is still greater than 0.7.

If the correlation is greater than or equal to 0.7, the correlation of consecutive frames after embedding the watermark is still within the threshold range. If the value is less than 0.7, the correlation is not within the threshold range. Then the shot segmentation algorithm is not consistent with the embedded video. So the shot segmentation algorithm is not suitable for the video watermarking algorithm.

As we can see from Fig. 6, although the correlation coefficients of the consecutive frames are different, they are greater than 0.7. However, the improved shot segmentation algorithm is effective for the video watermarking algorithm and it can still effectively segment the video after embedding the watermark.

Next then, the robustness of the video watermarking algorithm under various attacks is tested. This part mainly tests the influence of geometric attacks, compression attacks and several different recording attacks on watermark extraction. The geometric attacks include rotation, scaling, cropping and addition attacks. This paper applies C++ program language with OpenCV library to simulate geometric attacks. We used the software "Format Factory" to complete the compression

ISSN: 2454-4248
757 – 766

attack. In particular, the YUV format video after embedding watermark was encapsulated into an AVI video, and we converted it to video in MPEG-4, DivX or XviD format. Then we encoded it into AVI format and converted back to YUV format for watermark extraction. Finally, we tested the robustness of the proposed algorithm against recording attack. We recorded the video after embedding watermark by using the JVC (GY-HM200EC) camera at a distance of 2 meters from the 28-inch monitor. The accuracy rate of watermark detection in five single shot video under various attacks is shown in the following table:

TABLE III. THE ACCURACY RATE OF WATERMARK EXTRACTION UNDER VARIOUS ATTACKS

Attack Type	The accuracy rate					
	Seq1 (HD)	***Seq2 (HD)***	***Seq3 (HD)***	***Seq4 (HD)***	***Seq5 (HD)***	***Paper [10] (SD)***
Scaling down (1024×768)	100%	100%	100%	100%	100%	100%
Scaling down (800×600)	100%	100%	100%	100%	100%	100%
Addition 5%	100%	100%	100%	100%	100%	100%
Addition 10%	100%	100%	100%	100%	100%	100%
Cropping 5%	100%	100%	100%	100%	100%	100%
Cropping 10%	100%	100%	100%	100%	100%	100%
Rotate 3°	100%	100%	100%	100%	100%	100%
Rotate 5°	100%	100%	100%	100%	100%	100%
Compression (Mpeg-4)	100%	100%	100%	100%	100%	100%
Compression (Xvid)	100%	100%	100%	100%	100%	100%
Compression (Divx)	100%	100%	100%	100%	100%	100%
Camcorder recording1	100%	100%	100%	100%	100%	98.5%
Camcorder recording2	97.5%	97.5%	100%	100%	97.5%	97.5%
Camcorder recording3	95.0%	97.5%	95.0%	95.0%	97.5%	95.0%

Where the first five videos for high-definition single shot short video and paper [10] 's test video is the first five high-definition videos corresponding to the standard-definition video, the accuracy rate is an average accuracy rate of each video under various attacks. Although the correlation difference between the image and the watermark template under the geometric attacks discussed in the first part is large, the robustness test of the video after embedding watermark is good. Therefore, the watermark that is embedded to video by the proposed watermarking algorithm can be retained after the geometric attacks.

Camera recording ways of 1, 2, 3 are shown in Fig. 7 (a), Fig. 7 (b), Fig. 7 (c) respectively. As we can see, if the recording is done in a normal way, the accuracy rate of the watermark extraction is higher. However, if the recording way is done in other ways (like rotate, and addition), the accuracy rate of the watermark extraction is lower. But it is still higher than the accuracy of the watermark extraction of paper [10]'s method. For the video with multiple shots, the shot segmentation algorithm has a slight effect under the recording attacks. But there is still a high watermark extraction rate and detection rate.

Finally, we tested the watermark extraction rate and detection rate of video with multiple shots under various attacks. The results of the experiment are shown in the following table:

TABLE IV. THE DETECTION AND ACCURACY RATE OF WATERMARK EXTRACTION IN VIDEO WITH MULTIPLE SHOTS UNDER VARIOUS ATTACK

Table Head		The detection and accuracy rate					
		Scaling down (1024×768)	***Scaling down (800×600)***	***Addition 5%***	***Addition 10%***	***Cropping 5%***	***Cropping 10%***
Proposed (HD)	Det	100%	100%	100%	100%	100%	100%
	Acc	100%	100%	100%	100%	100%	100%
Paper [10] (SD)	Det	72.6 %	72.3 %	71.2 %	73.7 %	70.1 %	72.9 %
	Acc	100%	100%	100%	100%	100%	100%
Table Head		**The detection and accuracy rate**					
		Rotate 3°	**Rotate 5°**	**Compression**	**Camcorder recording1**	**Camcorder recording2**	**Camcorder recording3**
Proposed (HD)	Det	100%	100%	100%	82.1 %	80.8 %	76.2 %
	Acc	100%	100%	100%	99.5 %	98.0 %	94.5 %
Paper [10] (SD)	Det	68.6 %	70.6 %	71.2 %	67.8 %	65.2 %	66.9 %
	Acc	100%	100%	100%	98.5 %	97.5 %	95.0 %

Compression includes MPEG-4, Xvid, DivX compression, and the detection and accuracy rate of three kinds of compression is the same. So we merged to the one row - compression. As we can see from the table, the detection rate of video embedded watermarking by the paper [10]'s method is low because the paper [10]'s method only adapts to single shot video, and with smooth and still regions shot cannot be embedded effectively. In addition, recording attack is prone to the phenomenon of frame loss, frame fusion and other phenomena, and it leads to the decrease of the detection rate of watermark by the proposed algorithm. However, the detection rate of this algorithm under the recording attack is higher about 10%~30% than paper [10]'s method, and the detection rate can be guaranteed. Therefore, it is considered that the robustness of .the proposed algorithm is strong.

C. Analysis of Time Complexity

This part tests the time complexity of the algorithm. We use the five single frames of high-definition (1920×1080) shown in Fig.6 to test the time consumption of process which is a single frame to embed watermark. The comparison results of the proposed algorithm and Do et al.'s algorithm are shown in the following table:

(a) (b) (c)

Figure 7. three camcorder recording ways

TABLE V. THE COMPARISON OF TIME COMPLEXITY

Frame	Time complexity	
	Proposed	***Paper [10]***
Fig.5.(a)	563ms	1812ms
Fig.5.(b)	719ms	2094ms
Fig.5.(c)	703ms	1860ms
Fig.5.(d)	687ms	1875ms
Fig.5.(e)	625ms	1844ms

As we can see from Table 5, the time complexity of the proposed algorithm is three times lower than that of Do et al.'s algorithm and extracting process is similar as embedding process. So the embedding and extracting efficiency of the proposed algorithm is higher.

V. CONCLUSIONS

In this paper, we proposed an improved shot segmentation algorithm based on histogram correlation, calculated block-based histogram to generate watermark template, applied the human vision system based on variance to determine embedding areas and then to embed watermark in the temporal modulation way. Experimental results show that this algorithm can be applied to high-definition video and can effectively alleviate the problem of video flicker and can improve the detection rate of the watermark extraction, besides low time complexity, which is tested in the real recording environment. In addition, the proposed algorithm has good robustness to the camera and other accompanying attacks.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under Grant 61373020, Grant U1536102, and Grant U1536116, in part by China Mobile Research Fund Project (MCM20170407), the National Natural Science Foundation of China under Grant Nos. 61462018 and Guangzhou University mathematics Education Software Research Center Open Foundation under Grant Nos. LD16124X.

REFERENCES

[1] Zeng X, Chen Z, Chen M, and Xiong Z, "Reversible video watermarking based on inter-prediction error expansion," Journal of Computer-Aided Design & Computer Graphics, 2010, 22(5), pp. 871-878.
[2] Zeng X, Chen Z, Chen M, and Xiong Z, "Reversible video watermarking using motion estimation and prediction error expansion," Journal of Information Science & Engineering, 2011, 27(2), pp. 465-479.
[3] Vural C, and İbrahim Yıldırım, "Reversible video watermarking through recursive histogram modification," Multimedia Tools & Applications, 2017, 76, pp. 1-22.
[4] Zheng W, Li S D, Zhao X H, and Wang D S, "Histogram based watermarking algorithm resist to geometric attacks," Advanced Information Management, Communicates, Electronic and Automation Control Conference, IEEE, 2017, pp. 1438-1441.
[5] Zheng W, Liu W J, Li S D, Zhao XH, and Wang D S, "Histogram based watermarking algorithm robust to geometric attack with high embedding capacity," Advanced Information Management, Communicates, Electronic and Automation Control Conference, IEEE, 2017, pp. 1927-1933.
[6] Hu X, and Wang D, "A Histogram Based Watermarking Algorithm Robust to Geometric Distortions," International Conference on Electrical, Computer Engineering and Electronics, 2015, pp. 773-779.
[7] Alavianmehr M A, Rezaei M, Helfroush M S, and Tashk A, "A reversible data hiding scheme for video robust against H.264/AVC compression," International ISC Conference on Information Security and Cryptology, IEEE, 2013, pp.1-6.
[8] Chen C, Ni J, and Huang J, "Temporal statistic based video watermarking scheme robust against geometric attacks and frame dropping," International Workshop on Digital Watermarking, Springer-Verlag, 2009, pp. 81-95.
[9] Gao Q, Li Z, and Chen S, "A video dual watermarking algorithm against geometric attack based on integer wavelet and SIFT," International Conference on Cryptography, Security and Privacy, ACM, 2017, pp. 33-37.
[10] Do H, Choi D, Kim T, and Choi H, "Digital video watermarking based on histogram and temporal modulation and robust to camcorder recording," IEEE International Symposium on Signal Processing and Information Technology, IEEE, 2008, pp. 330-335.
[11] Gaj S, Rathore A K, Sur A, and Bora P K, "A robust watermarking scheme against frame blending and projection attacks," Multimedia Tools and Applications, 2017, 76(20), pp. 20755-20779.
[12] Leest A V, and Kalker T, "On digital cinema and watermarking," Proceedings of SPIE - The International Society for Optical Engineering, 2003, 5020, pp. 526-535.
[13] Wang D, Liu S, Luo X, and Li S, "A transcoding-resistant video watermarking algorithm based on corners and singular value decomposition," Telecommunication Systems, 2013, 54(3), pp. 359-371.