



## Article

# Adaptive Video Watermarking against Scaling Attacks Based on Quantization Index Modulation

Zhongze Lv <sup>1,2,†</sup> , Ying Huang <sup>1,†</sup>, Hu Guan <sup>1</sup>, Jie Liu <sup>1,\*</sup> , Shuwu Zhang <sup>1</sup> and Yang Zheng <sup>1</sup>

<sup>1</sup> Beijing Engineering Research Center of Digital Content Technology, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China; lvzhongze2017@ia.ac.cn (Z.L.); ying.huang@ia.ac.cn (Y.H.); hu.guan@ia.ac.cn (H.G.); shuwu.zhang@ia.ac.cn (S.Z.); yang.zheng@ia.ac.cn (Y.Z.)  
<sup>2</sup> School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China  
\* Correspondence: jie.liu@ia.ac.cn  
† These authors contributed equally to this work.

**Abstract:** Video watermarking plays a vital role in protecting the video copyright. The quantization-based methods are widely used in the existing watermarking algorithms, owing to their low computational complexity and completely blind extraction. However, most of them work poorly in resisting scaling attacks, by which the quantization value may fall outside the original quantization interval. For addressing this issue, an adaptive quantization index modulation method is proposed. The property that is associated with the ratio of the DC coefficient before and after scaling the video resolution motivates us to select the DC coefficient as the quantization value and set the size of the quantization interval by the video resolution to maintain the synchronization between them before and after scaling. Moreover, a strategy taking advantage of the high decoding reliability of the QR-Code is proposed to terminate the extraction in advance, and both the embedding and the extracting process are performed in the spatial domain, which all contribute to further enhance the execution efficiency. The experimental results show that our algorithm outperforms the state-of-the-art method in terms of imperceptibility, robustness, and computational cost.

**Keywords:** video watermarking; scaling attacks; adaptive; quantization index modulation



**Citation:** Lv, Z.; Huang, Y.; Guan, H.; Liu, J.; Zhang, S.; Zheng, Y. Adaptive Video Watermarking against Scaling Attacks Based on Quantization Index Modulation. *Electronics* **2021**, *10*, 1655. <https://doi.org/10.3390/electronics10141655>

Academic Editor: Frederic Ros

Received: 1 June 2021  
Accepted: 26 June 2021  
Published: 12 July 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The popularity of the Internet makes video infringement more rampant. Video watermarking [1–15], the process of embedding the copyright information called a watermark into the videos and then extracting the watermark to identify their ownership, can effectively protect the copyrights of the videos.

A video can be regarded as a series of images (frames). Embedding the watermark into these frames decoded by a video is the common way of the existing video watermarking process.

The existing video watermarking process embeds the watermark either in the spatial domain or in the transform domain. Spatial domain watermarking embeds the watermark by modifying the pixels of the original images directly. By contrast, the transform domain watermarking does it by adjusting the coefficients after some transformations. The former is highly efficient but less robust than the latter.

After determining the embedding domain, we should modify or analyze its coefficients to embed and extract the watermark. The quantization methods [7–10,16–21] are typical watermark embedding and extracting methods. It divides the reference axis into multiple index intervals, and each index interval corresponds to one component (0 or 1) of the watermark. The embedding quantizer quantizes the coefficients of the host signal into the index intervals corresponding to the component of the watermark to be embedded. The extracting quantizer analyzes the index interval where the coefficient falls to extract

the corresponding watermark. The quantization methods are popular for their low computational complexity and completely blind extraction, but they are generally inferior to scaling attacks. Due to the quantized coefficients changing with the signal after scaling, despite the index interval being fixed, the quantized coefficients may fall into different index intervals during embedding and extraction. In other words, scaling attacks cause a desynchronization between the quantized coefficients and the index intervals.

The watermarks can be embedded in either whole frames [3–8] or partial frames [9–14]. Obviously, the efficiency and the imperceptibility of the former are lower than that of the latter. For multiple watermarks extracted from these frames, there are two ways to determine the final watermark. One is providing the original watermark to determine the correctness of these extracted watermarks and return the watermark with the highest accuracy. However, this method will cause doubts about the authenticity of the final watermark, due the original watermark being known by the extractor. Moreover, the original watermark is usually blind in most extracting scenarios. Another way is to vote on all the extracted watermarks by bit, and the final watermark is composed of each bit with the highest vote. However, since there may be many watermarks extracted from the frames without the watermark, this disrupts the final result.

For solving the issues mentioned above, an adaptive quantization index modulation video watermarking algorithm against scaling attacks is presented, with the advantages of the DC coefficient [22,23] and the QRCode [3,24–27]. The innovations of this paper are as follows:

- (1) An adaptive quantization index modulation method is designed by analyzing the property of the DC coefficients. The method reaches the synchronization between the quantized coefficients and the index intervals before and after scaling.
- (2) A strategy to enhance the efficiency of the extraction process is proposed. It terminates the extraction process in advance, based on the high decoding reliability of the QRCode, reducing the execution time largely.

According to experiments and analysis, our algorithm is superior to the state-of-the-art algorithm in terms of imperceptibility, robustness, and computational cost.

The rest of this paper is organized as follows: Section 2 introduces the related work. Section 3 presents the adaptive quantization index modulation method proposed in this paper. Section 4 describes the strategy to enhance the efficiency of the extraction process. Section 5 shows the video watermarking embedding and extracting processes. The experimental results are given out in Section 6. Section 7 concludes the paper.

## 2. Related Work

Watermarking can be executed in a spatial domain [4,13,23,28,29] or a transform domain [3,5–12,14,15,18,19,21,22,24–27,30–35]. The spatial domain watermarking has high efficiency by modifying the pixels of the host images or frames directly to embed the watermark. In comparison, the transform domain watermarking modifies the coefficients obtained by transformation, such as the Discrete Cosine Transform (DCT) [8–10,18,19,21,22,24,26,30,31], Discrete Wavelet Transform (DWT) [7,11,12,14,25–27,32,33], Singular Value Decomposition (SVD) [6,12,25,26], Discrete Multiwavelet Transformation (DMT) [3], Contourlet Transform [6], Dual-Tree Complex Wavelet Transform (DT CWT) [5,15,34], and so on. They possess better imperceptibility and robustness, but their computational complexity is higher than that of the spatial domain watermarking. Furthermore, Su et al. [23] propose a watermarking which can modify the DC coefficients of DCT in the spatial domain directly. This algorithm can utilize the advantages of the spatial domain watermarking and the transform domain watermarking, with high efficiency and strong robustness.

Quantization index modulation (QIM) [16] is the first quantization method proposed in watermarking. The embedding quantizer quantizes the coefficients into disjoint index intervals, by the watermark to be embedding. The extracting quantizer analyzes the index interval in which the coefficient falls to obtain the corresponding watermark. Many watermarking algorithms [7–10,17] use QIM to embed and extract the watermark in the different

transformation domains. They adjust the index intervals according to the characteristics of the selected embedding domains and the embedding coefficients. However, their abilities to withstand scaling attacks are generally weak, because their index intervals are all the same during embedding and extracting, but the quantized coefficients will change with the scaling and they may fall into a different interval from the embedded one during extraction, which leads to incorrect extraction. An adaptive dither quantization index modulation model [18] is proposed by modifying the Waston's perceptual model, whose index intervals can be changed synchronously before and after scaling. This model cannot resist aspect ratio change attacks because it is only invariant to volumetric scaling, and its computational complexity is high. The differential quantization adaptive quantization threshold (DQAQT) [19] is an adaptive quantization method based on two-level DCT watermarking [30]. This method uses the stability of the difference between the adjacent components of the feature vectors in the two-level DCT domain and embeds the watermark by quantizing the difference. Due to the limitation of the embedding position, it has weak resistance to some large-scale scaling attacks. Meanwhile, this method has low efficiency because of performing two-level DCT, binary search, and so on.

Some algorithms select partial frames such as I-frames [9,10], keyframes [11,12], scene change frames [13,14], and so on, for improving efficiency and imperceptibility. Cedillo-Hernandez et al. [9] embed two watermark bits into each  $8 \times 8$  DCT block, which is located in the luminance component of I-frames. However, the watermarked I-frames cannot be obtained accurately when extracting. Sathya et al. [12] select the keyframes based on the Fibonacci sequence in each scene and apply a 2D wavelet transform to embed the watermark. The scene change is used by Li et al. [13] to choose suitable frames for embedding, so that the algorithm can resist the aimless frame dropping attacks. Masoumi et al. [14] detect the motion part of the video also by scene change analysis, and then a 3D wavelet transformation is applied to the detected part to embed the watermark. Whatever the case, if the original watermark is not known when extracting in these algorithms, they all need to integrate the whole watermarks extracted frame by frame to obtain the final watermark statistically, and the extraction accuracy may be affected by the frames without the watermark.

### 3. Adaptive QIM against Scaling Attacks in the Spatial Domain

To address the desynchronization between the index intervals and the quantized coefficients, an adaptive QIM is proposed in this section, based on the property that the ratio of the DC coefficients before and after scaling is related to the video resolution. It quantizes the DC coefficients by the quantization step (the quantization step is a parameter that controls the size of the index interval) changed with the video resolution, reaching the synchronization between the index intervals and the quantized coefficients, with strong robustness against scaling attacks.

Modifying the DC coefficients to embed the watermark is a good strategy, because of its strong robustness and high efficiency [22,23]. The DC coefficient denoted as  $F(0,0)$  can be obtained after the discrete cosine transform, which is calculated as:

$$F(0,0) = \frac{1}{\sqrt{ab}} \sum_{x=0}^{a-1} \sum_{y=0}^{b-1} f_{ij}(x,y) \quad (1)$$

where  $f_{ij}(x,y)$  is an image block with the size of  $a \times b$  in the  $i$ -th row and  $j$ -th column of the frame.

According to (1), the DC coefficient can be obtained in the spatial domain directly, instead of performing the discrete cosine transform. Furthermore, the formula of modifying the DC coefficient [23] is:

$$f'_{ij}(x,y) = \frac{1}{\sqrt{ab}} \Delta w + f_{ij}(x,y) \quad (2)$$

where  $f'_{ij}(x, y)$  is the watermarked image block and  $\Delta w$  is the modification of the DC coefficient. Moreover, the proposed adaptive QIM in this section is based on (2).

### 3.1. The Ratio of the DC Coefficients before and after Scaling

Assume that the original image block is  $I(x, y)$ , with the size of  $B_w \times B_h$ . Scale  $I(x, y)$  to  $G(x', y')$ , with the size of  $b_w \times b_h$ . The scaling factors are  $\alpha$  and  $\beta$  in the horizontal and vertical directions, respectively, so  $b_w = \alpha B_w$  and  $b_h = \beta B_h$ . The DC coefficients of the two blocks are denoted as  $DC_I$  and  $DC_G$ , separately, then calculated as follows:

$$DC_I = \frac{1}{\sqrt{B_w B_h}} \sum_{x=0}^{B_w-1} \sum_{y=0}^{B_h-1} I(x, y) \tag{3}$$

$$DC_G = \frac{1}{\sqrt{b_w b_h}} \sum_{x'=0}^{b_w-1} \sum_{y'=0}^{b_h-1} G(x', y') \tag{4}$$

The ratio between the two coefficients is denoted as  $\sigma$ , and

$$\sigma = \frac{DC_I}{DC_G} = \frac{\sqrt{b_w b_h} \sum_{x=0}^{B_w-1} \sum_{y=0}^{B_h-1} I(x, y)}{\sqrt{B_w B_h} \sum_{x'=0}^{b_w-1} \sum_{y'=0}^{b_h-1} G(x', y')} \tag{5}$$

According to (5), if the size of the image block before and after scaling is known,  $\sigma$  is determined by the ratio of the sum of pixel values before and after scaling. Define this ratio as  $\theta$  as follows:

$$\theta = \frac{\sum_{x=0}^{B_w-1} \sum_{y=0}^{B_h-1} I(x, y)}{\sum_{x'=0}^{b_w-1} \sum_{y'=0}^{b_h-1} G(x', y')} \tag{6}$$

Suppose that  $G(x', y')$  and  $I(x, y)$  are both continuous images, then  $G(x', y') = I\left(\frac{x}{\alpha}, \frac{y}{\beta}\right)$ , approximately. Furthermore,  $\theta$  can be calculated as:

$$\begin{aligned} \theta &= \frac{\int_0^{B_w} \int_0^{B_h} I(x, y) dx dy}{\int_0^{b_w} \int_0^{b_h} G(x', y') dx' dy'} = \frac{\int_0^{B_w} \int_0^{B_h} I(x, y) dx dy}{\alpha \beta \int_0^{\alpha B_w} \int_0^{\beta B_h} I\left(\frac{x}{\alpha}, \frac{y}{\beta}\right) d\frac{x}{\alpha} d\frac{y}{\beta}} \\ &= \frac{\int_0^{B_w} \int_0^{B_h} I(x, y) dx dy}{\alpha \beta \int_0^{B_w} \int_0^{B_h} I(x, y) dx dy} = \frac{1}{\alpha \beta} = \frac{B_w B_h}{b_w b_h} \end{aligned} \tag{7}$$

According to (5)–(7),

$$\sigma = \frac{\sqrt{b_w b_h}}{\sqrt{B_w B_h}} \theta = \frac{\sqrt{B_w B_h}}{\sqrt{b_w b_h}} \tag{8}$$

The images in the natural scene are all discrete, and there are interpolation operations during scaling. Hence, (8) is not an absolute equal relationship actually, but an approximate one shown as:

$$\sigma \approx \frac{\sqrt{B_w B_h}}{\sqrt{b_w b_h}} \tag{9}$$

The correctness of (9) is verified in Section 6.2.

Moreover, suppose that  $G(x', y')$  and  $I(x, y)$  are obtained by segmenting the Y component of a frame of the original video and the corresponding scaled video into  $W \times W$

blocks, respectively, with the resolution of  $M \times N$  and  $m \times n$  separately. Assuming that  $M$ ,  $N$ ,  $m$ , and  $n$  are all divided by  $W$ , therefore:

$$B_w B_h = \frac{MN}{WW} \quad (10)$$

$$b_w b_h = \frac{mn}{WW} \quad (11)$$

According to (9)–(11),  $\sigma$  is calculated as:

$$\sigma \approx \frac{\sqrt{MN}}{\sqrt{mn}} \quad (12)$$

### 3.2. Adaptive QIM Based on Video Resolution

Based on (12), when the DC coefficients are quantized, set the quantization step to  $\alpha\sqrt{MN}$  (where  $\alpha$  is a constant) during embedding, and  $\alpha\sqrt{mn}$  during extracting, then the synchronization between the index intervals and the quantized coefficients can be obtained before and after scaling. Then, an adaptive QIM against scaling attacks is obtained, which adjusts the quantization step based on the video resolution. The embedding formula of this method is as follows:

$$DC_{ij}^* = \begin{cases} (x - 0.5)\alpha\sqrt{MN}, & \text{if } (x + w_{ij}) \bmod 2 = 1 \\ (x + 0.5)\alpha\sqrt{MN}, & \text{if } (x + w_{ij}) \bmod 2 = 0 \end{cases} \quad (13)$$

where  $x = \text{round}\left(\frac{DC_{ij}}{\alpha\sqrt{MN}}\right)$ ;  $DC_{ij}$  is the DC coefficient of the image block located in the  $i$ -th row and  $j$ -th column of the frame for watermarking;  $\text{round}()$  is a rounding function;  $DC_{ij}^*$  is the value by embedding the watermark into  $DC_{ij}$ ;  $w_{ij}$  is the  $i$ -th row and  $j$ -th column of the watermark information; and  $w_{ij} \in \{0, 1\}$ ;  $\alpha$  is an adjustable parameter to balance imperceptibility and robustness. Then, the modification of the DC coefficient denoted as  $\Delta w$  can be calculated as:

$$\Delta w = DC_{ij}^* - DC_{ij} \quad (14)$$

According to (2) and (14), the embedding formula in the spatial domain is calculated as

$$f'_{ij}(x, y) = \frac{1}{\sqrt{ab}} \left( DC_{ij}^* - DC_{ij} \right) + f_{ij}(x, y) \quad (15)$$

Since (12) is an approximate result, rather than an absolute equal one, the watermark may not be extracted correctly by using  $\alpha\sqrt{mn}$  as the quantization step. However,  $\alpha\sqrt{mn}$  should be close to the quantization step which can extract the watermark accurately. Thus, it is possible to extract the watermark correctly, after several fine tunings near  $\alpha\sqrt{mn}$ . Based on this, a numerical interval is determined based on  $\alpha\sqrt{mn}$ , and the boundary of the interval is determined by experiments. The numerical points in the interval are sampled at a certain distance, which are utilized as the quantization steps to extract the watermark iteratively. The extracting formula is as follows:

$$E'_{ij} = \begin{cases} 1, & \text{if } x' \bmod 2 = 1 \\ 0, & \text{if } x' \bmod 2 = 0 \end{cases} \quad (16)$$

where  $x' = \text{floor}\left(\frac{DC'_{ij}}{\Delta}\right)$ ;  $\text{floor}()$  is the function of rounding down;  $DC'_{ij}$  is the DC coefficient of the image block located in the  $i$ -th row and  $j$ -th column of the watermarked frame;  $\Delta$  represents the value sampled by  $step$  in an interval denoted as  $[\alpha\sqrt{mn} - \varepsilon, \alpha\sqrt{mn} + \varepsilon]$ ;  $\varepsilon$  and  $step$  are both empirical values; and  $E'_{ij}$  is the watermark information extracted from the image block, locating in the  $i$ -th row and  $j$ -th column of a frame.

#### 4. A Strategy to Enhance the Efficiency of the Extraction Process

Typical video watermarking embeds a string or an image as the watermark. If the original watermark is not known during extraction, the watermarking must integrate all the watermarks extracted from the whole frames that may contain the watermark to obtain the final watermark. In this situation, the execution efficiency is low, due to performing the extraction operation many times. Meanwhile, the final watermark may be influenced by the frames which do not have the watermark. To solve these issues, a strategy to enhance the efficiency of the extraction process is proposed, by taking advantage of the QRCode.

Some existing watermarking schemes already embed the QRCode as the watermark [3,24–27], but none of them consider its high decoding reliability, but just the strong error correction. However, by making good use of this characteristic, the video watermarking performance can be enhanced significantly. Hence, we mainly focus on the high decoding reliability, differing from the previous work.

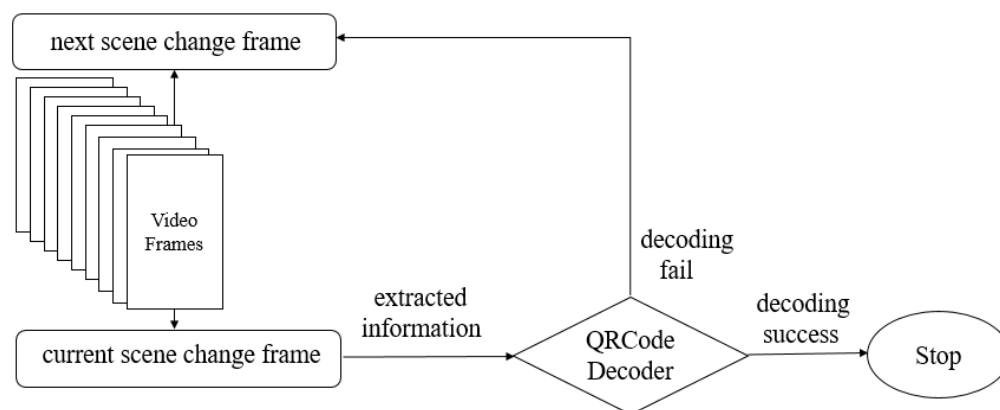
##### 4.1. Terminating the Extraction Process in Advance

The high decoding reliability means that when a QRCode can be decoded successfully, then it has been encoded by the original watermark string. According to this, by embedding the QRCode into frames, a strategy to enhance the extraction efficiency is proposed: during frame by frame extraction, if the QRCode extracted in the current frame can be successfully decoded by the decoder, then it is regarded as the watermark of the entire video and the extraction process is terminated. The extraction process will no longer be performed in the rest frames, and the execution times on the frames are reduced.

The above proposed strategy can be applied to the whole frames, and we only select the scene change frames as an example. The scene change frame is the first frame in a scene of a video. It varies quickly when the video is played and embedding the watermark into it is not easily perceived by human eyes. Meanwhile, it can be located quickly during extraction due to its robustness against most attacks, and the extraction efficiency is improved. The correlation coefficient between the histograms of the Y components of two consecutive frames will not exceed a certain empirical value if a scene change frame occurs, so it is calculated to judge whether the current frame is a scene change frame. Assume that  $H_f$  and  $H_{f+1}$  are the histograms of the previous and current frame, respectively, then the correlation coefficient denoted as  $d(H_f, H_{f+1})$  between them is calculated as:

$$d(H_f, H_{f+1}) = \frac{\text{cov}(H_f, H_{f+1})}{\sqrt{D(H_f)D(H_{f+1})}} \quad (17)$$

where  $\text{cov}(H_f, H_{f+1})$  is the covariance between  $H_f$  and  $H_{f+1}$ ;  $D(H_f)$  is the variance of  $H_f$ ; and  $D(H_{f+1})$  is the variance of  $H_{f+1}$ . If  $d(H_f, H_{f+1})$  does not exceed a threshold denoted as *Thre*, then the current frame is regarded as a scene change frame. The flow chart of the whole process is shown in Figure 1.

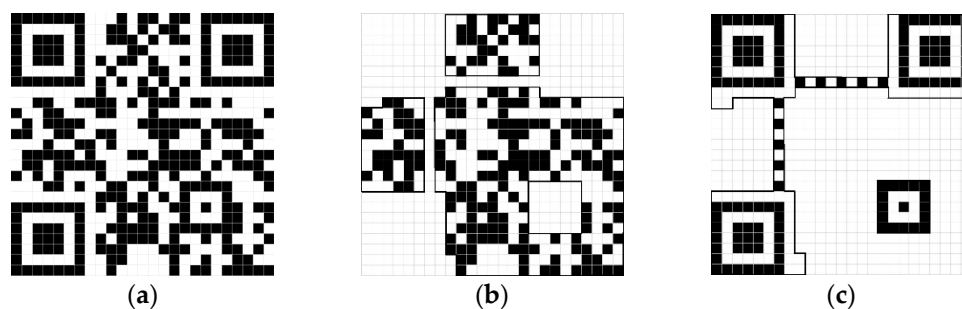


**Figure 1.** Terminating the extraction process in advance.

#### 4.2. Reducing the Amount of the Embedded Data

When the embedding strength is constant, the imperceptibility will decrease as the embedding amount of the watermark increases. Embedding a complete QRCode with a large amount of data will injure the imperceptibility. Thus, based on the encoding characteristic of the QRCode [24], this section reduces the amount of the embedded data for optimizing the strategy proposed in Section 4.1, to obtain better imperceptibility under the same robustness.

The QRCode owns four error correction levels (the L-level, the M-Level, the Q-level, and the H-level), with the approximate correction amounts of 7%, 15%, 25%, and 30%, respectively. Furthermore, we select the H-level QRCode to reach the maximum error correction ability. A complete QRCode can be divided into two parts: the excess regular area, and the important data area. The excess regular area is an ordered pattern, and the important data area mainly contains the valuable information. Based on the encoding characteristic, if the size and error correction level of the QRCode are known, the excess regular area does not vary with the different information to be encoded [24]. Hence, we can embed the important data area instead of the original QRCode, and the size and the error correction level are saved for inferring the excess regular area during extracting. For example, a  $25 \times 25$  QRCode and its two areas are shown in Figure 2, the amount of embedded data can be reduced by 38% approximately, with only embedding the important area.



**Figure 2.** A  $25 \times 25$  QRCode and its two parts. (a) The  $25 \times 25$  QRCode; (b) Excess regular area; (c) Important data area.

In summary, only embedding the important area can reduce the amount of the data to be embedded greatly, enhancing the imperceptibility under the same robustness. Although the size and the error correction level of the original embedded QRCode need to be known during extraction, the existing watermarking schemes also must limit the size of the original watermark.



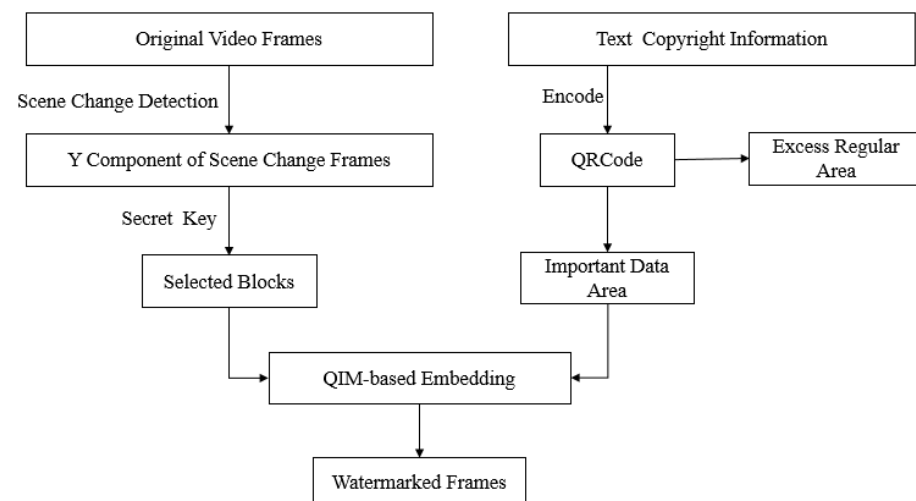
## 5. Video Watermarking Scheme

This section mainly describes the video watermarking embedding and extracting processes, with the adaptive QIM proposed in Section 3.2, and the strategy presented in Section 4.1.

### 5.1. Video Watermarking Embedding Process

The flow chart of the embedding process is shown in Figure 3 and the detailed steps are presented below:

- Step 1:** Encode the watermark information into a  $W \times W$  QRCode; divide the QRCode into the excess regular area and the important data area; denote the important data area as  $E$ .
- Step 2:** Obtain the Y component of the scene change frame, according to (17); enlarge the Y component to the size of  $M \times N$ , with a small scaling factor, so that  $M$  and  $N$  both can be divided by  $W$ .
- Step 3:** Segment the Y component into  $W \times W$  blocks; use a secret key to select some blocks whose number is equal to the number of the data in  $E$ .
- Step 4:** For each selected block, embed one-bit information of  $E$ , according to (15). When  $E$  has been embedded in all selected blocks, rescale the frame to the original size to obtain the watermarked frame. Then, the watermark will be embedded into the next scene change frame until the last frame of the video is decoded.



**Figure 3.** The video watermarking embedding process.

### 5.2. Video Watermarking Extracting Process

The flow chart of the extracting process is shown in Figure 4, and the detailed steps are presented below:

- Step 1:** Obtain the Y component of the scene change frame, according to (17); enlarge the Y component to the size of  $m \times n$ , with a small scaling factor, so that  $m$  and  $n$  both can be divided by  $W$ .
- Step 2:** Segment the Y component into  $W \times W$  blocks; select the corresponding image blocks to extract the watermark by using the same secret key as the embedding process.
- Step 3:** Extract the watermark to obtain the corresponding important data area denoted as  $E'$ , according to (16).
- Step 4:** Combine the excess regular area with  $E'$  to reconstruct the QRCode, and if the QRCode can be decoded successfully, terminate the extracting process.
- Step 5:** If the watermark is not extracted correctly in the current scene change frame, it will be extracted from the next scene change frame.



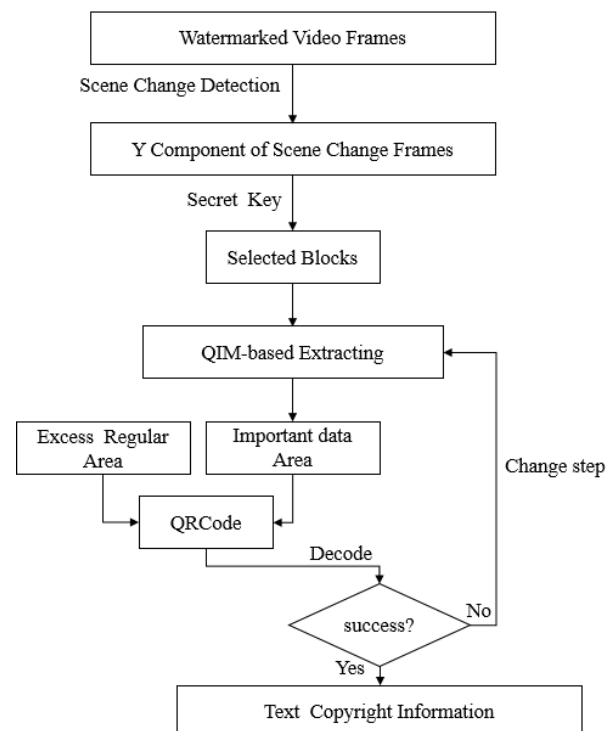


Figure 4. The video watermarking extracting process.

## 6. Experiments and Analysis

This section mainly evaluates the performance of the proposed video watermarking. Section 6.1 introduces the data sets, the experimental environment, and the parameter settings. Section 6.2 verifies the correctness of (9). Section 6.3 discusses the imperceptibility. Section 6.4 shows the comparisons of robustness between our algorithm and the differential quantization adaptive quantization threshold (DQAQT) [19]. Section 6.5 compares the computational cost.

### 6.1. Experimental Setup

Two data sets denoted as “Video 1080P” and “Video 720P” are created, with the resolution of 1080P ( $1920 \times 1080$ ) and 720P ( $1280 \times 720$ ), respectively. They all contain 50 videos in the mp4 format, with a duration ranging from 90–360 s and a frame rate ranging from 23.98–30 frames per second. Furthermore, the number of the videos in the two data sets is large enough in the field of the video watermarking. All 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 OpenCV2.4.9, Python 3.6, and FFmpeg 2.1. *Thre* is set to 0.6.  $\epsilon$  is set to 0.5. *step* is set to 0.1.

DQAQT is applied to the video watermarking as the comparison algorithm. It also modifies the coefficients in the DCT domain, with good robustness against a variety of attacks. For making the comparisons fair, the two algorithms choose the same scene change frames to embed the watermark: DQAQT embeds a 16-byte string; our algorithm embeds a  $25 \times 25$  QRCode encoded from the same string. Moreover, in DQAQT, to satisfy blind extraction (i.e., do not know the original string during extraction), the results on all frames should be integrated, to obtain the final watermark, statistically.

### 6.2. Verifying the Property of the DC Coefficients before and after Scaling

One hundred  $80 \times 60$  image blocks are randomly selected in the natural scene for verifying the property of the DC coefficients before and after scaling shown in (9). Their size is close to the size of the blocks obtained by segmenting the Y components in our algorithm. Four interpolation methods with different factors are selected to perform the

scaling. The four interpolation methods are nearest neighbor interpolation (denoted as *NEAR*), bilinear interpolation (denoted as *LINE*), bicubic interpolation (denoted as *CUBI*), and pixel relation resampling (denoted as *AREA*). Furthermore,  $\sqrt{B_w B_h} / \sqrt{b_w b_h}$  is denoted as *RATI*.

Shrink the image blocks with three scaling factors: 0.2, 0.4, and 0.6. The comparisons between  $\sigma$  and *RATI* under shrinking are shown in Figure 5a.

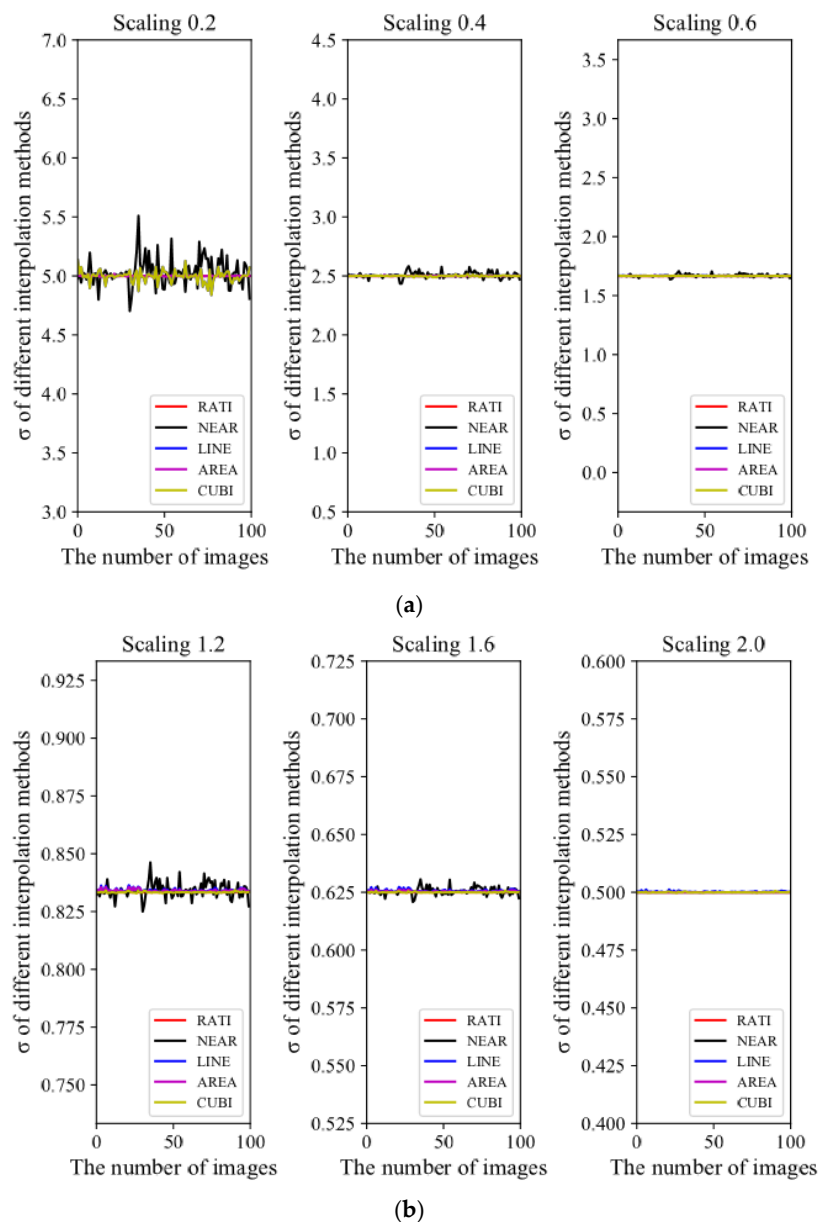


Figure 5. The comparisons between  $\sigma$  and *RATI* under scaling. (a) Shrinking; (b) Enlarging.

Enlarge the image blocks with three scaling factors: 1.2, 1.6, and 2.0. The comparisons between  $\sigma$  and *RATI* under enlarging are shown in Figure 5b.

According to Figure 5, the larger the scaling factor is, the closer *RATI* is to  $\sigma$ , regardless of shrinking or enlarging. Hence, (9) is correct unless the scaling factor is too small.

### 6.3. Evaluation of Imperceptibility

This subsection evaluates the imperceptibility of the two algorithms, based on the mean peak signal to noise ratio (MPSNR) [2]. The larger the MPSNR is, the better the imperceptibility is. The imperceptibility of our algorithm is determined by  $\alpha$ . When  $\alpha$  starts

from 0.10 to 0.17, with an interval of 0.01, and the averages of MPSNR on the two data sets are shown in Table 1. If the value of MPSNR is more than 40 dB, the imperceptibility is perceived as good. Thus,  $\alpha$  is set to 0.16 on Videos 1080P and 0.15 on Videos 720P, and the corresponding MPSNRs are 41.99 dB and 41.63 dB, respectively. For making the comparisons fair, the MPSNR of DQAQT is set to 41.74 dB on Videos 1080P and 41.41 dB on Videos 720P, by adjusting its parameters.

**Table 1.** The averages of MPSNR (dB) with different  $\alpha$ .

Data Sets	0.10	0.11	0.12	0.13	0.14	0.15	0.16	0.17
Video 1080P	45.11	44.42	43.89	43.41	42.97	42.43	41.99	41.57
Video 720P	43.89	43.17	42.88	42.51	42.13	41.63	41.29	40.91

#### 6.4. Comparisons of Robustness

The capabilities of the two algorithms to recover the embedded watermark in various scenarios are compared in the subsection, including geometric attacks, combined attacks, and network media propagation attacks.

The byte error rate (BER) is used to evaluate the robustness, which is calculated as:

$$BER = \frac{Num(errorBytes)}{Num(totalBytes)} \quad (18)$$

where  $Num(errorBytes)$  is the number of bytes inconsistent between the extracted watermark and the original watermark, and  $Num(totalBytes)$  is the number of bytes of the original watermark. The smaller the BER is, the better the robustness is.

##### 6.4.1. Robustness against Geometric Attacks

This subsection shows the comparisons of the robustness against four geometric attacks: resizing, scaling, aspect ratio change, and shielding. The results on the two data sets are shown in Table 2, where ‘-’ denotes that the watermark cannot be extracted. From the table, we can see that the robustness of our algorithm against geometric attacks is better than that of DQAQT. Many frames that can satisfy (12) in our algorithm, even if shrinking the videos with a very small scaling factor. Meanwhile, our algorithm combines strong error tolerance of the QRCode and iterative extraction, improving the accuracy of the extraction. DQAQT is restricted to the sixth coefficient selected in each block for watermark embedding. After scaling or resizing to a small size, the size of each block is less than six, and the sixth coefficient cannot be obtained again. Furthermore, although shielding removes part of information ineluctably, our algorithm can still recover the watermark correctly due to the stability of the DC coefficient and the strong error tolerance of the QRCode.

##### 6.4.2. Robustness against Combined Geometric Attacks

This subsection evaluates the performance of the two algorithms for a set of combined geometric attacks including resizing and scaling, aspect ratio change and resizing, resizing and shielding, and scaling and shielding, and the results are shown in Table 3. It reveals that the proposed algorithm has better robustness than DQAQT even if two types of attacks occur at the same time. It is not surprising to get such results, because our algorithm shows sufficient advantages only when dealing with a certain type of attacks.

##### 6.4.3. Robustness against Network Media Propagation Attacks

In addition to the above attacks, this subsection focuses on the robustness against network media propagation attacks in realistic scenarios, considering the video transmission in the complex network environment.

**Table 2.** The comparisons of robustness against geometric attacks.

Data Sets		Videos 1080P		Videos 720P	
Attack Types	Parameters	Proposed (%)	DQAQT (%)	Proposed (%)	DQAQT (%)
Resizing	0.1	0	-	72	-
	0.15	0	4	0	-
	0.2	0	0	0	2
	0.3	0	0	0	0
Scaling	0.1	0	-	-	-
	0.15	0	-	22	-
	0.4	0	0	0	0
	1.5	0	0	0	0
	2.0	0	0	0	0
Aspect ratio change	0.15 × 0.1	0	-	28	-
	1.5 × 1.2	0	0	0	0
	0.1 × 2	0	-	0	-
	1.5 × 0.15	0	-	0	-
Shielding	25%	0	0	0	0
	36%	0	11.97	0	12.47
	49%	0	13.06	0	13.28

**Table 3.** The comparisons of robustness against combined geometric attacks.

Data Sets		Videos 1080P		Videos 720P	
Attack Types	Parameters	Proposed (%)	DQAQT (%)	Proposed (%)	DQAQT (%)
Resizing and Scaling	0.1 & 0.9	0	-	84	-
	0.15 & 0.9	0	0.45	0	-
	0.2 & 0.9	0	0	0	0.52
Aspect ratio Change and Resizing	0.15 × 0.1 & 0.9	88	-	0	-
	0.1 × 1.5 & 1.5	0	-	12	-
	1.5 × 0.15 & 0.9	0	-	24	-
Resizing and Shielding	0.1 & 10%	0	-	84	-
	0.15 & 10%	0	1.11	0	-
	0.2 & 10%	0	-	0	0.88
Scaling and Shielding	0.7 & 36%	0	12.54	0	13.16
	0.6 & 36%	0	12.67	0	13.45
	0.5 & 36%	0	13.27	0	12.86

Select ten videos denoted as V1-V10 from Videos 1080P randomly and apply the two algorithms to embed the watermark into them. Upload them to three network media platform websites (MicroBlog [35], Zhihu [36] and Bilibili [37]), and then download these videos to test robustness. MicroBlog and Bilibili can download the videos with four resolutions: 1080P (1920 × 1080), 720P (1280 × 720), 480P (850 × 480), and 360P (640 × 360). Zhihu can download the videos with two resolutions: 720P (1280 × 720) and 480P (848 × 478). During the whole process, various attacks (such as transcoding, compression, and scaling) are faced on the videos at the same time. Furthermore, in order not to infringe the copyrights of the videos, all the videos were deleted immediately after the whole operation, ensuring that they would not be transmitted to the network. The comparisons of imperceptibility and robustness against network media propagation attacks between the two algorithms are shown in Tables 4 and 5, respectively. The results of our algorithm are better, based on the strong fault tolerance of QRCode and the stability of the DC coefficients. Our algorithm is more suitable for real application.

**Table 4.** The comparisons of MPSNR (dB) on the ten selected videos.

Algorithms	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
Proposed	42.2	42.5	42.5	42.8	41.9	42.0	42.4	42.1	42.3	42.0
DQAQT	41.7	41.8	41.9	41.8	41.8	41.8	41.8	41.7	41.7	41.7

**Table 5.** The comparisons of robustness against network media propagation attacks in realistic scenarios.

Websites	Download Resolution	Proposed (%)	DQAQT (%)
MicroBlog	1080P	0.00	0.00
	720P	0.00	1.33
	480P	0.00	2.28
	360P	0.00	12.50
Zhihu	720P	0.00	1.02
	480P	0.00	1.02
Bilibili	1080P	0.00	0.00
	720P	0.00	0.00
	480P	0.00	0.86
	360P	0.00	21.17

### 6.5. Computational Cost

This subsection mainly compares the computational cost of our algorithm and DQAQT, with average embedding and extracting time without attacks.

The average embedding time of the two algorithms are shown in Table 6. The embedding time of our algorithm is more than ten times shorter, due to modifying the DC coefficients in the spatial domain directly.

**Table 6.** The comparisons of the average embedding time per frame.

Data Sets	Proposed(s)	DQAQT(s)
Videos 1080P	0.053	0.677
Videos 720P	0.024	0.308

Table 7 shows the average extracting time without attacks of the two algorithms on the two data sets. The time of the proposed algorithm is much less because it uses the strategy presented in Section 2 to terminate the extraction process in advance and extracts the watermark in the spatial domain directly.

**Table 7.** The comparisons of the average extracting time without attacks.

Data Sets	Proposed (s)	DQAQT (s)
Videos 1080P	3.401	82.794
Videos 720P	1.525	35.877

## 7. Conclusions

By analyzing the ratio of the DC coefficients before and after scaling, we conclude that it is approximately equal to the ratio of the video resolutions before and after scaling. Based on this property, an adaptive QIM method in the spatial domain is designed, which solves the problem that the traditional quantitation method has a weak ability to resist the scaling attacks. Taking advantage of the high decoding reliability of the QRCode, a strategy to enhance the efficiency of the extraction process is proposed by terminating the process in advance. Furthermore, we optimize this strategy by reducing the amount of the embedded data based on the encoding characteristic of the QRCode. According to these, an adaptive QIM video watermarking based on video resolution is obtained. The

algorithm can effectively resist geometric attacks, combined geometric attacks, and also network media propagation attacks with good imperceptibility in realistic scenarios. The algorithm has high efficiency and can be applied to practical application scenes.

In future work, we will improve the robustness of the algorithm to resist more geometric attacks, so that it can be applied to more complex realistic scenarios.

**Author Contributions:** Conceptualization, Z.L. and Y.H.; methodology, Z.L. and Y.H.; software, Z.L.; validation, Z.L., Y.H. and H.G.; formal analysis, Z.L., Y.H. and H.G.; investigation, Z.L., Y.H. and H.G.; resources, S.Z.; data curation, J.L.; writing, Z.L. and Y.H.; original draft preparation, Z.L., Y.H., H.G. and Y.Z.; writing—review and editing, Z.L., Y.H., H.G., J.L. and Y.Z.; visualization, Z.L. and Y.H.; supervision, Y.H. and H.G.; project administration, S.Z.; funding acquisition, S.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Key R&D Program of China (2019YFF0302800) 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.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data is available from the corresponding author upon reasonable request.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## References

- Asikuzzaman, M.; Pickering, M.R. An overview of digital video watermarking. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *28*, 2131–2153. [[CrossRef](#)]
- Yu, X.; Wang, C.; Zhou, X. A survey on robust video watermarking algorithms for copyright protection. *Appl. Sci.* **2018**, *10*, 1891. [[CrossRef](#)]
- Ketcham, M.; Ganokratanaa, T. The evolutionary computation video watermarking using quick response code based on discrete multiwavelet transformation. In *Recent Advances in Information and Communication Technology*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 113–123.
- Rigoni, R.; Freitas, P.G.; Farias, M.C. Detecting tampering in audio-visual content using qim watermarking. *Inf. Sci.* **2016**, *328*, 127–143. [[CrossRef](#)]
- Esfahani, R.; Akhaee, A.; Norouzi, Z. A fast video watermarking algorithm using dual tree complex wavelet transform. *Multimed. Tools Appl.* **2019**, *78*, 16159–16175. [[CrossRef](#)]
- Barani, M.J.; Ayubi, P.; Valandar, M.Y.; Irani, B.Y. A blind video watermarking algorithm robust to lossy video compression attacks based on generalized newton complex map and contourlet transform. *Multimed. Tools Appl.* **2020**, *79*, 2127–2159. [[CrossRef](#)]
- Yassin, N.I.; Salem, N.M.; El Adawy, M.I. Qim blind video watermarking scheme based on wavelet transform and principal component analysis. *Alex. Eng. J.* **2014**, *53*, 833–842. [[CrossRef](#)]
- Cedillo-Hernandez, A.; Cedillo-Hernandez, M.; Garcia-Vazquez, M.; Nakano-Miyatake, M.; Perez-Meana, A. Transcoding resilient video watermarking scheme based on spatio-temporal hvs and dct. *J. Vis. Signal Process.* **2014**, *97*, 40–54. [[CrossRef](#)]
- Cedillo-Hernandez, A.; Cedillo-Hernandez, M.; Miyatake, M.N.; Meana, H.P. A spatiotemporal saliency-modulated jnd profile applied to video watermarking. *J. Vis. Commun. Image Represent.* **2018**, *52*, 106–117. [[CrossRef](#)]
- Zhao, H.; Dai, Q.; Ren, J.; Wei, W.; Xiao, Y.; Li, C. Robust information hiding in low-resolution videos with quantization index modulation in dct-cs domain. *Multimed. Tools Appl.* **2018**, *77*, 18827–18847. [[CrossRef](#)]
- Ponni alias Sathya, S.; Ramakrishnan, S. Non-redundant frame identification and keyframe selection in dwt-pac domain for authentication of video. *IET Image Process.* **2019**, *14*, 366–375. [[CrossRef](#)]
- Ponni alias Sathya, S.; Ramakrishnan, S. Fibonacci based key frame selection and scrambling for video watermarking in dwt-svd domain. *Wirel. Pers. Commun.* **2018**, *102*, 2011–2031. [[CrossRef](#)]
- Li, X.; Wang, X.; Yang, W.; Wang, X. A robust video watermarking scheme to scalable recompression and transcoding. In *Proceedings of the 2016 6th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, Beijing, China, 17–19 June 2016; pp. 257–260.
- Masoumi, M.; Amiri, S. A blind scene-based watermarking for video copyright protection. *AEU-Int. J. Electron. Commun.* **2013**, *67*, 528–535. [[CrossRef](#)]
- Goria, L.E.; Pickering, M.R.; Nasiopoulos, P.; Ward, R.K. A video watermarking scheme based on the dual-tree complex wavelet transform. *IEEE Trans. Inf. Forensic Secur.* **2008**, *3*, 466–474.



16. Chen, B.; Wornell, G.W. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. *IEEE Trans. Inf. Theory* **2001**, *47*, 1423–1443. [[CrossRef](#)]
17. Fan, D.; Wang, Y.; Zhu, C. A blind watermarking algorithm based on adaptive quantization in contourlet domain. *Multimed. Tools Appl.* **2019**, *78*, 8981–8995. [[CrossRef](#)]
18. Li, Q.; Cox, I.J. Using perceptual models to improve fidelity and provide resistance to volumetric scaling for quantization index modulation watermarking. *IEEE Trans. Inf. Forensic Secur.* **2007**, *2*, 127–139. [[CrossRef](#)]
19. Huang, Y.; Niu, B.; Guan, H.; Zhang, S. Enhancing image watermarking with adaptive embedding parameter and psnr guarantee. *IEEE Trans. Multimedia.* **2019**, *21*, 2447–2460. [[CrossRef](#)]
20. Sadeghi, M.; Toosi, R.; Akhaee, M.A. Blind gain invariant image watermarking using random projection approach. *Signal Process.* **2019**, *163*, 213–224. [[CrossRef](#)]
21. Ali, M.; Ahn, C.W.; Pant, M.; Kumar, S.; Singh, M.K.; Saini, D. An optimized digital watermarking scheme based on invariant DC Coefficients in spatial Domain. *Electronics* **2020**, *9*, 1428. [[CrossRef](#)]
22. Huang, J.; Shi, Y.Q.; Shi, Y. Embedding image watermarks in dc components. *IEEE Trans. Circuits Syst. Video Technol.* **2000**, *10*, 974–979. [[CrossRef](#)]
23. Su, Q.; Chen, B. Robust color image watermarking technique in the spatial domain. *Soft Comput.* **2018**, *22*, 91–106. [[CrossRef](#)]
24. Chen, J.H.; Chen, W.Y.; Chen, C.H. Identification recovery scheme using quick response (qr) code and watermarking technique. *Appl. Math. Inf. Sci.* **2014**, *8*, 585. [[CrossRef](#)]
25. Patvardhan, C.; Kumar, P.; Lakshmi, C.V. Effective color image watermarking scheme using ycbcr color space and qr code. *Multimed. Tools Appl.* **2018**, *77*, 12655–12677. [[CrossRef](#)]
26. Rosales-Roldan, L.; Chao, J.; Nakano-Miyatake, M.; Perez-Meana, H. Color image ownership protection based on spectral domain watermarking using qr codes and qim. *Multimed. Tools Appl.* **2018**, *77*, 16031–116052. [[CrossRef](#)]
27. Thulasidharan, P.P.; Nair, M.S. Qr code based blind digital image watermarking with attack detection code. *AEU-Int. J. Electron. Commun.* **2015**, *69*, 1074–1084. [[CrossRef](#)]
28. Xiang, S.; Kim, H.J.; Huang, J. Invariant image watermarking based on statistical features in the low-frequency domain. *IEEE Trans. Circuits Syst.* **2008**, *18*, 777–790.
29. Hua, G.; Xiang, Y.; Zhang, L.Y. Informed histogram-based watermarking. *IEEE Signal Process. Lett.* **2020**, *27*, 236–240. [[CrossRef](#)]
30. Guan, H.; Zeng, Z.; Liu, J.; Zhang, S. A novel robust digital image watermarking algorithm based on two-level dct. In Proceedings of the 2014 International Conference on Information Science, Electronics and Electrical Engineering, Sapporo, Japan, 26–28 April 2014; pp. 1804–1809.
31. Mareen, H.; Van Kets, N.; Lambert, P.; Van Wallendael, G. Fast Fallback Watermark Detection Using Perceptual Hashes. *Electronics* **2021**, *10*, 1155. [[CrossRef](#)]
32. Rajani, D.; Kumar, P.R. An optimized blind watermarking scheme based on principal component analysis in redundant discrete wavelet domain. *Electronics* **2021**, *10*, 1155. [[CrossRef](#)]
33. Zhou, X.; Ma, Y.; Zhang, Q.; Mohammed, M.A.; Damaševičius, R. A reversible watermarking system for medical color images: Balancing capacity, imperceptibility, and robustness. *Electronics* **2021**, *10*, 1024. [[CrossRef](#)]
34. Liu, J.; Rao, Y.; Huang, Y. Complex wavelet-based image watermarking with the human visual saliency model. *Electronics* **2019**, *8*, 1462. [[CrossRef](#)]
35. Microblog. Available online: <http://www.weibo.com/> (accessed on 27 April 2021).
36. Zhihu. Available online: <http://www.zhihu.com/> (accessed on 27 April 2021).
37. Bilibili. Available online: <http://www.bilibili.com/> (accessed on 27 April 2021).