Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2015, Article ID 789536, 11 pages
http://dx.doi.org/10.1155/2015/789536

Research Article

A Real-Time Video Watermarking Algorithm for Authentication of Small-Business Wireless Surveillance Networks

Shaohui Liu,[1] Daniel Bo-Wei Chen,[2] Long Gong,[1] Wen Ji,[3] and Sanghyun Seo[4]

[1] School of Computer Science and Technology, Harbin Institute of Technology, No. 92 West Dazhi Street, Harbin 150001, China
[2] Department of Electrical Engineering, National Cheng Kung University, Road 1, Tainan 70101, Taiwan
[3] Institute of Computing Technology, Chinese Academy of Sciences, No. 6 Kexueyuan South Road, Zhongguancun, Beijing 100190, China
[4] Electronics and Telecommunications Research Institute, 138 Gajeongno, Yuseong-gu, Daejeon 305-700, Republic of Korea

Correspondence should be addressed to Daniel Bo-Wei Chen; dennisbwc@gmail.com

Received 13 December 2014; Revised 21 January 2015; Accepted 2 March 2015

Academic Editor: Naveen Chilamkurti

In some business applications, all kinds of cameras sensors are employed in a distributed way to capture videos for different tasks such as surveillance. Once some illegal actions happen, then somebody or some organization wants to forge or replace some surveillance video clips to destroy evidences or obtain illegal profits. How to authenticate the genuineness and integrity of the source video or trace the source of a video information leak becomes a growing requirement in these small businesses. Fortunately, video watermark just provides an effective technology to resolve this issue. This paper proposes a real-time video watermarking scheme for MPEG, where firstly exploits fast scenes segmentation to original video sequence and adaptively selects appropriate scenes to be embedded. Furthermore, visual model is utilized to modulate watermark strength. Watermarks are embedded by adjusting the number of *bit1* in the bitstreams through changing level of run-level pairs. Experiment results show little loss of video quality and also exhibit excellent robustness against many attacks. As watermark is directly detected in bitstreams domain, real-time detection becomes a reality. In addition, the embedding strategy guarantees that the bit rate is not increased and the experiments also validate it.

1. Introduction

The rapid development of internet technologies has extremely accelerated the speed of information exchange and extended the channel of information exchange. Moreover as we know "a picture is worth a thousand words"; audio and video have semantically much richer than just text or/and images. Thus, applications of image and video have growing from the begging of this century. Now with the development of multimedia technology and the increasing of internet bandwidth, people prefer audio and video with most expressive form of media to just single media such as text or images. In recent years, there have been an increasing number of available websites providing video servers, for example, video downloads, online video, and video sharing services such as YouTube [1–3]. Particularly the development of digital imaging and network technologies make cheap digital video production and convenience and fast video transmission possible. Compared with few years ago, nowadays social networks, such as, Facebook, Twitter, QQ, and Weixin, all provide video sharing and playing services [4–6]. However, the wide dissemination and the feature of making perfect copy easily make privacy protection, authentication and access control become a growing concern, the video watermarking provides a potential measure to these problem and has been a research hotspot in recent years [7, 8].

Let us take video surveillance as an example. Nowadays, video surveillance is very popular everywhere. A security supervisor should guarantee that the surveillance videos recorded everyday must not be changed by other people or employees.

This task is not an easy task for each security supervisor. But once the forging happens, it will lead to inestimable loss. This is actually happening in a real case. Some employee stole money from a company and replaced the original surveillance video with a fake surveillance video to cheat the security supervisor. Actually, some researchers thought that using video surveillance as evidence in court is absolute reliability [9]. Although surveillance video can be actualized by witnesses acquainted with the video subject, the genuineness of video must be guaranteed by some technologies. Otherwise, the forged video will make many troubles. Thus, how to achieve the destination is urgent. Fortunately, video watermarking or video data hiding provides a very promising solution for this. Particularly in some privacy preserving environments, there are some very interesting attempts [10, 11].

Generally speaking, invisibility, robustness, and real-time processing are the major challenges of video watermarking technology [12]. One video watermarking technology always wants to keep better invisibility and stronger robustness and less processing time. However these three features conflict with each other. Hence, a good video watermarking algorithm will achieve the best tradeoff among these features under some constraints of the algorithm's application environment. Invisibility means that the distortion caused by watermark embedding is imperceptible by human eyes. Robustness requires that the watermark algorithm can resist variety of intentional and unintentional attacks. Because the watermark can be essentially looked at as the noise embedded by the watermarking algorithm in original signal, the much stronger robustness will lead to the weaker invisibility of watermarking algorithm. The real-time processing requires the lower time-complexity so that the watermark embedding and extraction do not delay remarkably the normal video operations, for example, play, and download. Otherwise, the video watermark will degrade the user experience. Döerr and Dugelay [13] deemed that real-time is the big challenge of video watermarking and thought that there are two ways to improve the real-time feature; one is to lower the algorithm's complexity and the other is to transfer the computation burden to the video provider or watermark embedding side; thereafter the complexity of client or detection side is decreased.

In the earlier period of watermarking, invisibility and robustness are heavily emphasized and the real-time is neglected. For example, Swanson et al. [14] proposed a video watermark algorithm based on human visual system model and scene segmentation and achieved the good tradeoff between invisibility and robustness by adjusting the wavelet coefficients of video frames; however the computational complexity is too large. Niu et al. [15] proposed to apply wavelet transform to watermark and original video frames and then used error correction code to improve the robustness of the method. But the algorithm is time-consuming. Nowadays, due to the development of internet bandwidth, digital video applications have been widely used in daily living; VOD (video on command), real-time interactive video, online live, and so on request the real-time processing. Even normal video applications also demand low-delay, hence it is more important than ever to real-time processing of watermarking.

Real-time video watermarking always embeds the watermark in compressed domain to avoid the computational burden in video coding. Hartung and Girod [16] proposed a watermarking algorithm in compressed domain by using spread-spectrum method and they used the drift-compensation measure to improve the robustness of the watermark. In [17], Langelaar and Lagendijk proposed an algorithm called Differential Energy Watermarking (DEW) which embeds the watermark code by adjusting the coefficients' energy relationship between two DCT coefficients blocks. If the energy relationship does not satisfy the required relationship to embed one bit, one can adjust the relationship to meet the required relationship by removing some high frequency DCT coefficients in one DCT block. The algorithm is performed in the low bit-rate environment. These two methods embed the watermark in DCT domain, and the watermark extraction requires that the bit-stream should be decoded by entropy-decoding first and then inverse quantization. Hence, they are complicated from a computational standpoint.

For the real-time characteristic, Langelaar et al. [18] further proposed another watermarking algorithm based on changing the level value of run-level pairs in entropy coding stage of video coding. Because the method only changed the LSB bit of the level value, it obtained good invisibility and real-time characteristics at the cost of robustness. Similarly, Lu et al. [19] also proposed a watermarking algorithm in VLC domain, where it embedded the watermark bit into video by adjusting the mean value of all level values in one whole macroblock (MB); however, it does not provide a valid control on quality degradation and its robustness to time-synchronization attack is not good.

In recent years, real-time video watermark in compressed domain has become one of main tendencies of watermarking technologies; many works have been proposed. Ye et al. [20] proposed an improved adaptive real-time video watermarking algorithm, where it is based on visual characteristic to select suitable watermarking positions, and watermark bits are embedded into the video streams with dedundant styles through exchanging the EQSP (equal quantization step position). Lu et al. [21] proposed a real-time frame-dependent video watermarking in VLC domain. Roy et al. proposed a hardware implementation for video authentication [22].

Those methods usually considered some aspects of invisibility, robustness, and real-time. However, watermark in compressed domain embeds the watermark in DCT domain or VLC domain in video bits-stream. This paper fully considers these three aspects of video watermarking and proposes a new watermarking algorithm which can achieve the real-time detection and processing. The main contributions of the paper are summarized as follows.

First, fast video scene segmentation is applied to choose those scenes with abundant texture and larger variance as candidate scenes which will be embedded watermark.

Second, once those candidate scenes are determined then a visual model is used to determine the largest amount of changeable values of one (Run, Level) pair in order to guarantee the invisibility of watermark.

The watermark bits are embedded into the video by changing the value of level and then adjusting the relationship between the number of *bit1* in two group subblocks in each macroblock. Hence, the watermark detection can be easily done by counting the number of *bit1*.

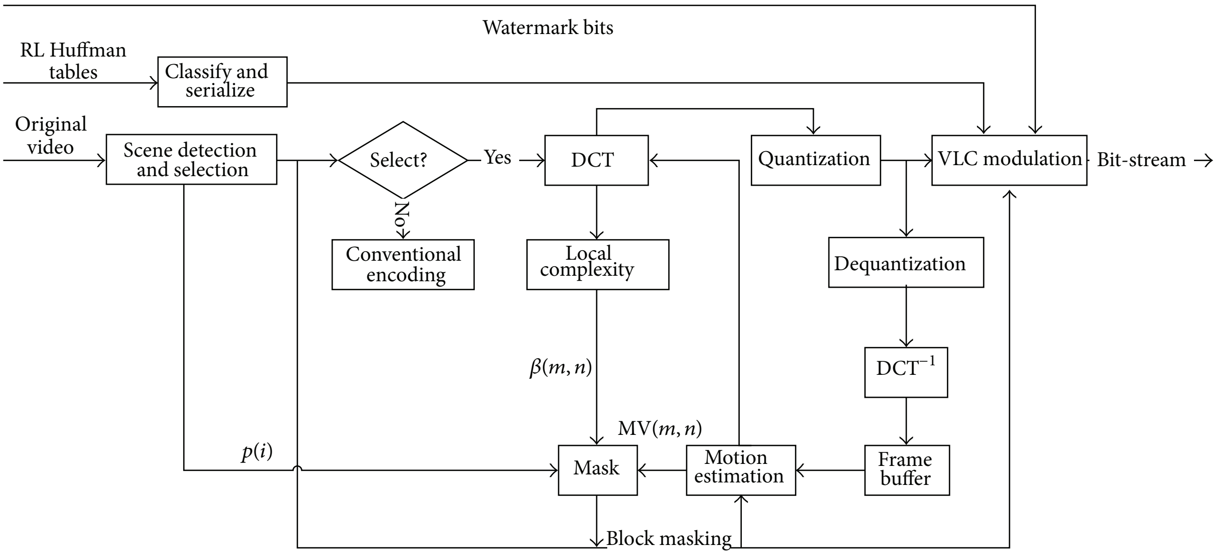


Figure 1: The framework of watermark embedding.

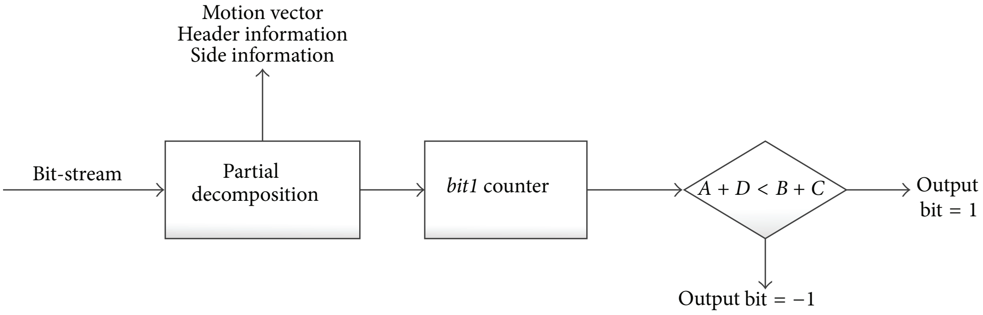


Figure 2: The extraction framework of watermark.

In order to resist time-synchronized and collusion attacks, the same watermark information is embedded repetitively into each frame in each candidate's scenes and different watermark information is embedded into different candidate scenes, respectively.

The rest of the paper is organized as follows. The second section overviews the algorithm and then introduces each component in detail. The third section is the experiments and discussion. The final is the conclusion.

2. The Proposed Watermark Algorithm

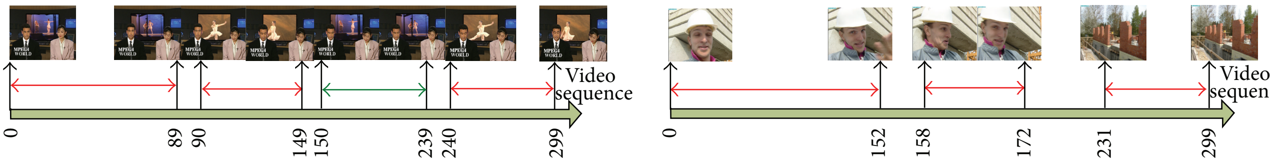
In order to achieve real-time applications, the watermark embedding can be done in video coding; of course it can be done in compressed video. It depends on your applications. The extraction of watermark can be finished in decoding process or independently computed by extracting algorithm.

The framework of proposed algorithm is shown in Figure 1. This figure takes the basic block-based video coding framework as basis and emphasizes the four key components of proposed algorithm: the detection and selection of candidate scenes, the partition and ordering of Huffman tables, visual modeling, and watermark embedding. Where $p(i)$ indicates the global embedding strength of the ith frame in the scene, $\beta(m,n)$ denotes the local embedding strength of the 8 by 8 subblock located at (m,n) in ith frame. In this paper, the same watermark bits are embedded into each frame in every candidate scene. Moreover, one bit watermark is embedded into each chosen macroblock.

Figure 2 is the extraction illustration. The extraction process is relatively simple. The bit-stream first flows into a filter which can filter the motion vector, head information, and side information; then the number of *bit1* in groups A and D is compared with that in groups B and C to determine the extracted bit (the four subblocks A, B, C, and D are arranged from left to right and from top to bottom).

In the following section, each key component is given in detail.

2.1. Segmentation and Section of Candidate Scenes. Scene is defined as a series of frames which are taken by one shot (or several shots with slow movement). A meaningful scene cannot be deleted completely without loss of semantic meaning; hence repeatedly embedding the same watermark in one scene can provide the robustness against varied time-synchronized attacks (such as averaging, deleting, and regrouping frames).



(a) News sequence

(b) Foreman sequence

Figure 3: The result of scene segmentation.

Although there are some matured scene segmentation methods [23–25], a fast and appropriate scene segmentation considering the real-time requirement of video watermarking is proposed. It is well known that DC coefficients in each 8×8 subblock mean the average luminance of the block; all DC coefficients in one frame can denote the main information of that frame; thus the relationship of DC coefficients in adjacent frames is used to segment the scene.

Let $D(i, m)$ be the mth DC coefficient in the ith frame; then the changed amount $\text{Var}(i)$ of the DC coefficient of the ith frame against that of the previous frame can be defined as

$$\text{Var}(i) = \frac{1}{M}\sum_{m}\left(D(i,m) - D(i-1,m)\right)^2. \tag{1}$$

Then calculate the growth rate of changed amount as

$$\alpha(i) = \frac{\text{Var}(i) - \text{Var}(i-1)}{\min\left(\text{Var}(i), \text{Var}(i-1)\right)}. \tag{2}$$

In order to decrease the effect caused by adjacent frames with still or nearly still scenes which will lead to some errors in scene segmentation, the changed amount $\text{Var}(i)$ will bound as $\text{Var}(i) = \overline{\text{Var}(i)}/3$ if $\text{Var}(i) < \overline{\text{Var}(i)}/3$.

The first frame of one scene has dramatically changed in comparison with the last frame in its immediate previous scene. Namely, the changed amount of DC in this case will change much greater than that in adjacent frames in the same scenes. This means that the larger α. In the similar way, the changed amount of DC in the second frame in one scene will much less than that in the first frame which means that the less α. Hence, the segmentation of scenes can be observed as the alternate seeking procedure for the start frame and the end frame of scene:

$$\begin{aligned}\text{Star Frame} &= \left\{i - 1 \,\middle|\, \alpha(i) < -\xi\right| \mid \text{Var}(i) < \beta_1\}, \\ \text{End Frame} &= \left\{i - 1 \,\middle|\, \alpha(i) > \xi\right| \mid \text{Var}(i) > \beta_2\}.\end{aligned} \tag{3}$$

In our experiments, $\xi = 2$, $\beta_1 = 30$, and $\beta_2 = 300$. Figure 2 shows the result of scene segmentation for foreman and news test sequence. The gap between scenes is the interspace without semantic meaning.

According to the characteristic of HVS (human visual system), scenes with high complexity and high variance between frames have high redundancy and thus they have better invisibility than those scenes with low complexity and low variance when the same amount watermark information are embedded into these two class scenes. For finding the appropriate candidate scenes quickly, a parameter p which indicates the degree of appropriateness for embedding watermark is proposed as

$$\begin{aligned}p = {} & \left(\frac{1}{M \times (N-1)}\sum_{m}\sum_{n}\left(D(i,m,n+1) - D(i,m,n)\right)^2\right. \\ & \left. + \frac{1}{(M-1) \times N}\sum_{m}\sum_{n}\left(D(i,m+1,n) - D(i,m,n)\right)^2\right) \\ & \times \text{Var}(i),\end{aligned} \tag{4}$$

where $D(i, m, n)$ indicates the DC of subblock located at (m, n) in the ith frame. i is the second frame in that scene. The parameter p in each scene is calculated firstly and then is compared with a predefined threshold to determine whether the scene is a candidate scene or not. It is observed from (4) that the parameter p is calculated as the product of the energy of gradient of DC coefficient of the second frame and the changed amount of the first frame in that scene. In Figure 3, the red maker indicates that the scene is chosen as a candidate scene and the green maker means that the scene is not chosen according to formula (4).

2.2. Partitioning and Ordering of Huffman Tables.

AC coefficients are coded by run coding in most MPEG video compression standards. The two-tuples in run coding can be denoted as $(\mathit{Run}, \mathit{Level})$, where *Level* indicates the nonzero value of the DCT coefficient after current quantization and *Run* indicates the number of continuous 0 before the coefficient. Each $(\mathit{Run}, \mathit{Level})$ corresponds to one Huffman table. This mapping can be written as $(r, l) \rightarrow h$; then all Huffman tables can be represented as a set, $Q = \{((r, l), h)\}$. According to the r and the bit length of Huffman coding, all Huffman tables can be partitioned as

$$Q_{R,H} = \{((r,l), h) \mid r = R, |h| = H\}. \tag{5}$$

After sorting partitioned sets in ascending order by the value of l, the number of *bit1* in each h is calculated. In order to improve the robustness of the watermark, the concept of best appropriate embedding code table is defined as $((r, l), h)$, if and only if $h(r, l-1)$, $h(r, l)$, and $h(r, l+1)$ have the same number of *bit1*.

For example, MPEG-4 has different run-level code mechanisms for different frame type: inter, intra, interlast, and intralast. An example of partitioned results is shown in Box 1.

Actually, this partition is just the equivalent partition in set theory. A similar idea has been proposed in [26].

```
(1 2 2 1 0) (2F 4 40) (3 15 6 3 0) (4 17 7 40) (5 1F 8 5 0) (6 25 9 3 0) (7 24 9 2 0) (8 21 10 2 0) (9 20 10 1 0)
(10 7 11 3 0) (11 6 11 2 0) (12 20 11 1 0)
(1 6 3 2 1) (2 14 6 2 0) (3 1E 8 40) (4 F 10 4 0) (5 21 11 2 0) (6 50 12 2 1)
(1 E 4 3 0) (2 1D 8 4 0) (3 E 10 3 0) (4 51 12 3 1)
(1 D 5 3 1) (2 23 9 3 1) (3 D 10 3 1)
(1 C 5 2 1) (2 22 9 2 0) (3 52 12 3 0)
(1 B 5 3 0) (2C 10 2 0) (3 53 12 4 0)
(1 13 6 3 1) (2B 10 3 1) (3 54 12 3 1)
(1 12 6 2 1) (2A 10 2 1)
(1 11 6 2 1) (29 10 21)
                                             inter
(1 2 21 0) (2 6 3 2 0) (3 F 4 40) (4D 5 30) (5 C 5 2 0) (6 15 6 30) (7 13 6 3 0) (8 12 6 20) (9 177 40)
(1 E 4 3 0) (2 146 2 0) (3 16 7 3 0) (4 1C8 3 0)(5 20 9 1 0) (6 1F9 5 0) (7 D 10 3 0)
(1 B 5 3 1) (2 15 7 3 0) (3 1E9 40) (4 C 10 2 0) (5 56 12 4 0)
(1 11 6 2 0) (2 1B 8 40) (3 1D9 4 0) (4 B 10 3 0)
(1 106 1 0) (2 22 9 2 0) (3 A 10 2 1)
(1 D 6 3 1) (2 1C 9 3 0) (3 8 10 1 0)
(1 12 7 2 0) (2 1B 9 40) (3 54 12 3 0)
(1 14 7 2 0) (2 1A 9 3 0) (3 57 12 50)
(1 19 8 3 0) (29 10 20)
                                             intra
```

Box 1: The partial result of human table partition.

2.3. Visual Model. Human eyes have different sensitivities to changed amounts of coefficients at different positions in one frame. Hence, a sophisticated position choosing criteria is designed according to HVS to mask watermark information as natural noise and a good invisibility of watermark is benefited from this mechanism.

The invisibility is closely related with the embedding strength. In this paper, the complexity of scenes, spatial complexity, and time-complexity are considered to determine the embedding strength. The complexity of scenes is represented by the parameter p in Section 2.1. In spatial complexity of images, most people think texture area is more appropriate for embedding watermark than smooth and edge areas. Thus the energy of high frequency components of its DCT coefficients in a video frame describes the spatial complexity of the frame in some sense. Suppose $\beta(m,n)$ denotes the spatial complexity of the (m,n) subblock, and A denotes the high frequency area of DCT coefficients. Then $\beta(m,n)$ is defined as

$$\beta(m,n) = \sum_{(i,j)\in A} \left|\mathrm{DCT}(i,j)\right|. \tag{6}$$

In addition, human eyes are sensitive to motion part in videos. Thus motion factor is considered to adjust the embedding strength. Consider the motion information in video coding, namely, motion vector $\mathrm{MV}(m,n)$. $\mathrm{MV}(m,n)$ is a constant if the prediction mode in some block is intramode. Then the embedding strength of subblock (m,n) is represented as

$$\gamma(m,n) = \frac{p \times \beta(m,n)}{\mathrm{MV}(m,n)}, \tag{7}$$

where $\mathrm{DCT}(m,n,i,j)$ denotes the DCT coefficient at the ith row and jth column of the subblock (m,n), and S_r, the range of search of corresponding level value, can be represented as

$$S_r = \frac{(\gamma(m,n)/\mathrm{step}(i,j))}{r}, \tag{8}$$

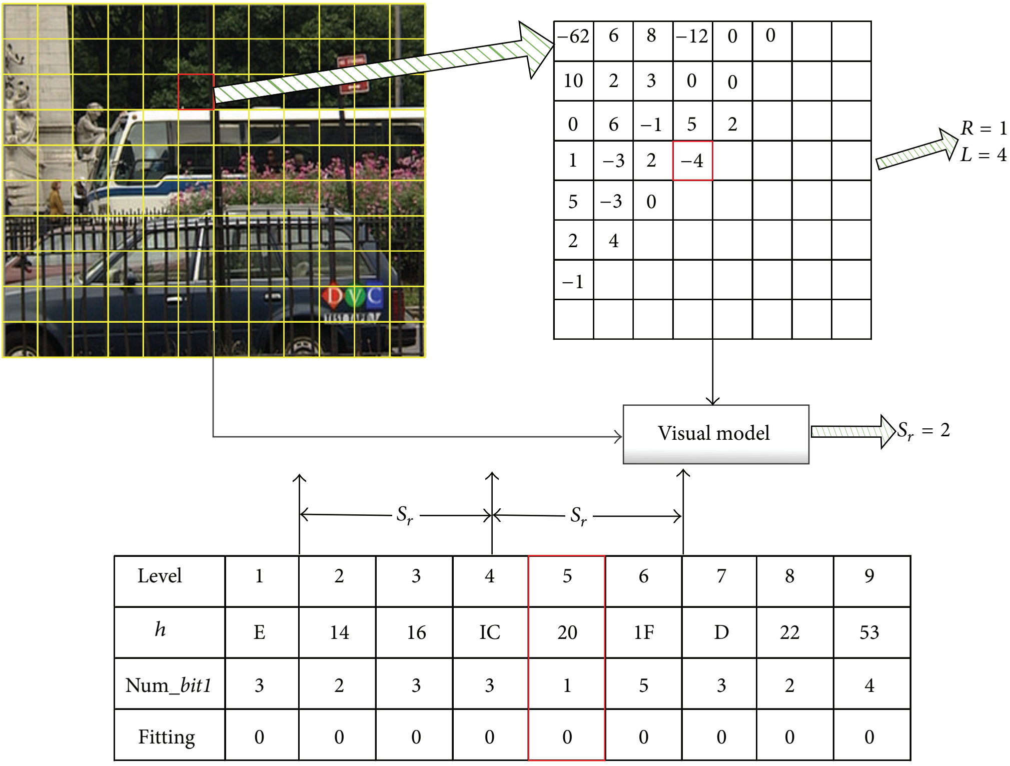
where $\mathrm{step}(i,j)$ indicates the quantization step and r denotes the scaling factor by experience.

2.4. Watermark Embedding. Selecting the appropriate candidate scenes based on scene segmentation, then the embedding strength is determined by HVS model. The same watermark is embedded into each frame in every scene. Following a detailed embedding algorithm to embed one bit information in one chosen microblock is introduced.

For confidentiality of watermark information, a key K is chosen to generate a pseudo random sequence S_N which is used as the positions of candidate microblock (MB). The embedding space is luminance space. Each MB is divided into 4 subblocks A, B, C, and D which are regrouped as (A, D) and (B, C). Then adjusting the number of *bit1* in these two bitsstreams embeds the watermark bit according to

$$\begin{aligned} &\mathit{Num\_bit}1\,(A + D) < \mathit{Num\_bit}1\,(B + C) \quad w(k) = 1, \\ &\mathit{Num\_bit}1\,(A + D) > \mathit{Num\_bit}1\,(B + C) \quad w(k) = -1. \end{aligned} \tag{9}$$

w is a bipolar watermark signal. $w(k)$ is the kth bit. $\mathit{Num\_bit}1(A+D)$ denotes the number of *bit1* in group (A, D). It is found in experiments that the number of *bit1* in (A, D) and (B, C) is sometimes too different to adjust the run value to embed the watermark bit. In order to solve the problem, those MBs whose differences of the number of *bit1* between



−62	6	8	−12	0	0		
10	2	3	0	0			
0	6	−1	5	2			
1	−3	2	−4				
5	−3	0					
2	4						
−1							

Level	1	2	3	4	5	6	7	8	9
h	E	14	16	IC	20	1F	D	22	53
Num_*bit1*	3	2	3	3	1	5	3	2	4
Fitting	0	0	0	0	0	0	0	0	0

R = 1 for intratable

FIGURE 4: The embedding of watermark bits.

(A, D) and (B, C) falling into some range are used to embed watermark bits as 10:

$$\begin{aligned} &-N < Num\_bit1\,(A+D) - Num\_bit1\,(B+C) < 0 \\ &\qquad\qquad w(k) = 1, \\ &0 < Num\_bit1\,(A+D) - Num\_bit1\,(B+C) < N \\ &\qquad\qquad w(k) = -1, \\ &\qquad \text{otherwise} \quad \text{not embedding.} \end{aligned} \tag{10}$$

It is noted that N should be chosen carefully to decrease the distortion caused by embedding. Actually, at least two criteria should be considered. The first is the number of macroblocks which satisfy (10) should be large as much as possible. This condition can guarantee the higher capacity of proposed algorithm. The second condition is that the value of N cannot be too large to avoid the difficulty of adjusting the number of *bit1*. Thus these two conditions conflict with each other. Experiments have verified this fact. Table 1 shows the distribution of MB in intra- and interframes. Considering these two conditions, N is fixed as 30 in the experiments. Extensive experiments also indicate that the distribution of the difference of bit1 is similar to Gaussian distribution with zero mean.

In the following we introduce how to change the number of *bit1* in specific bit-stream. Taking $w(k) = 1$ as an example, the embedding algorithm decreases the number of *bit1* in group (A, D) to embed bits while keeping the compressed file size unchanged greatly. Suppose (R, L) is a run-level pair in subblock A, L corresponds to the location (i, j) of DCT coefficients. The corresponding search range S_r (indicates the largest unnoticeable changed amount of L) of L can be calculated in (8). Then seeking the Huffman tables obtained in (5) to get a new L' to substitute L

TABLE 1: The percentage of macroblocks satisfying (10) under different N.

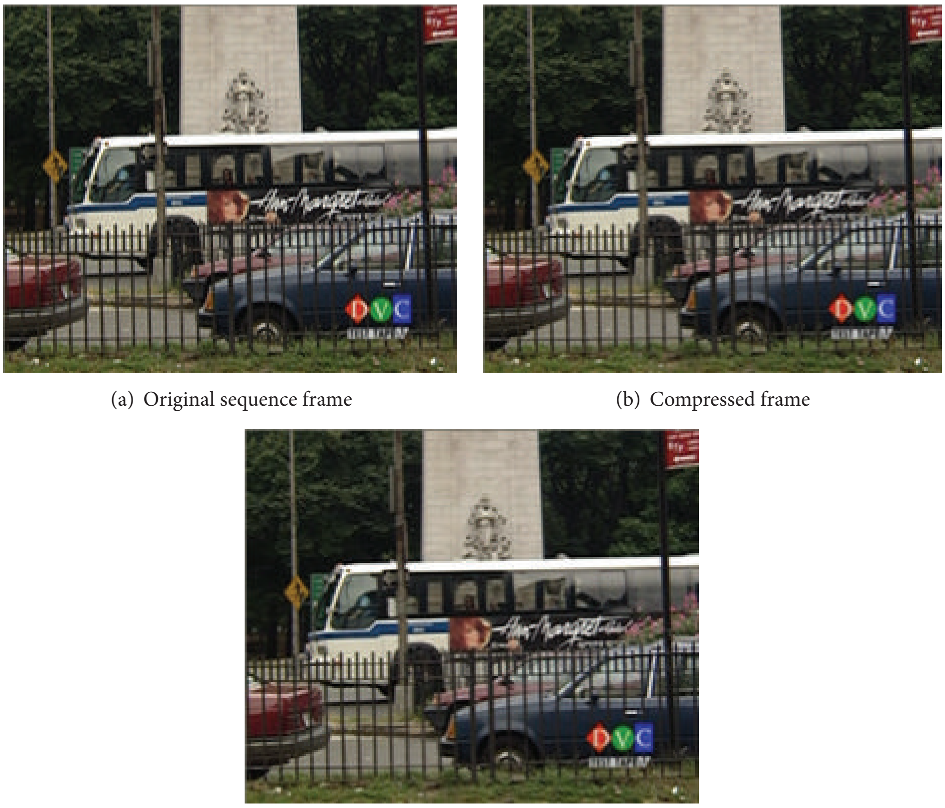
Type\\N	10	20	30	40	50
Intra	0.43	0.63	0.76	0.84	0.90
Inter	0.55	0.73	0.83	0.89	0.93

$$L' = \mathop{\arg\min}_{\substack{|l-L|<S_r \\ |h(l)|=|h(L)| \\ fitting\ code}} (Num\_bit1\,(h(l)))\,. \tag{11}$$

It is observed that when $|h(l)| = |h(L)|$, the search range is very small; hence removing the constrain condition will magnify the search range:

$$L' = \mathop{\arg\min}_{|l-L|<S_r} (Num\_bit1\,(h(l)))\,. \tag{12}$$

In Figure 4, the specific embedding framework is shown. In each selected level, there is an interval determined by S_r in which the appropriate candidate level is searched.



(a) Original sequence frame

(b) Compressed frame

(c) The watermarked frame

Figure 5: The 50th frame in Bus test sequence.

2.5. Watermark Extracting. Compared with the embedding, the detection and extraction of watermark are straightforward. The specific extraction is illustrated as in Figure 2. The relationship of the number of *bit1* in (A, D) and (B, C) is used to determine the watermark bit.

3. Experiments

As mentioned earlier, invisibility, robustness, and real-time are the three important factors in video watermarking. Following experiments on these three aspects will be conducted extensively to validate the performance. The video codec is MPEG-4; the coding and decoding codec is provided by Project Mayo of DivX Advance Research Center. The test sequences are the standard CIF sequences.

3.1. Invisibility. Video watermarking requires the watermarking process does not degrade dramatically the perceptual quality of video. In our experiments bus_cif sequence is used to embed the watermark. Figure 5 shows the 50th frame where the compression bit-rate is 7.12 Mbps.

From the images in Figure 5 human eyes cannot see any distortion caused by watermark embedding. Besides subjective tests, PSNR is used to measure the objective quality. The corresponding PSNR is plotted in Figure 6. The average PSNR is larger than 40 dB which indicates the good video quality. Moreover, the difference of PSNR values between compressed video with and without watermark embedding is very small. It indicates that the watermark embedding has negligible effect on the quality of compressed video.

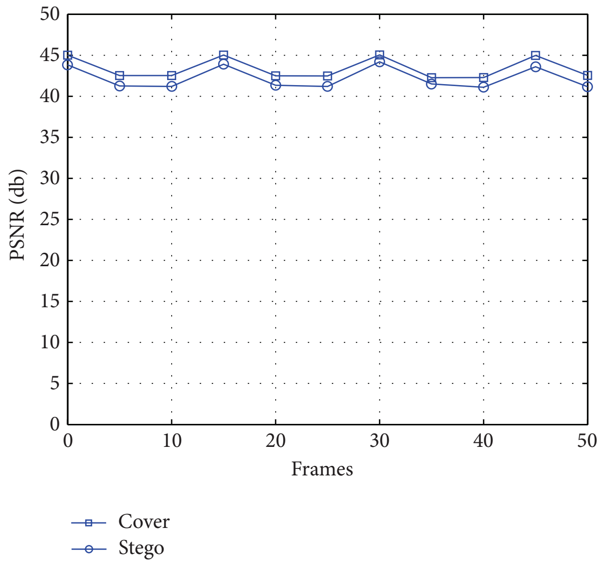


Figure 6: The PSNR difference between compressed video with and without watermark embedding.

Table 2: The detection of watermarked scenes.

	SC	WSC	DSC	ESC
Bus	1	1	1	0
Foreman	3	3	3	0
News	4	4	4	0
Basketball	2	2	2	0
Football	3	2	2	0
Glassgow	19	13	12	0
Toto	19	16	16	0

3.2. Robustness. Robustness is the capability of resisting all kinds of attacks. Attacks include active and passive attacks. For example, the quality degradation caused by noise signal is a type of passive attack. And removing the watermark by deleting some frames is a typical active attack.

In experiments, compressed video with watermark is decompressed and conducts all kinds of attacks; finally it is recompressed again. Hence, recompression attack is subject to other attacks. In the following we discuss the robustness of proposed algorithm from the different aspects.

3.2.1. The Detection Performance without Any Attacks. $w(k) \in \{1, -1\}$ indicates the kth watermark bit. $\widehat{w}(k)$ indicates the extracted kth bit. And the bit correct rate is defined as BCR (bit correct rate):

$$\mathrm{BCR} = \frac{\sum_{k=0}^{K} w(k)\,\widehat{w}(k)}{K}. \tag{13}$$

If BCR $>$ 0.9, then the frame is deemed to contain a watermark; if one frame in some scene has been judged to have a watermark, then the scene is deemed to contain a watermark. In experiments, different video sequences are used to test the performance. The number of scenes (SC), watermarked scenes (WSC), detected watermarked scenes (DSC), and error detected watermarked scenes (ESC) is displayed in Table 2.

The toto sequence is a mixed video sequence formed by many different video sequences and it includes 3979 frames. From this table, most of watermarked scenes can be detected without any attacks. BCR of each frame in one scene also indicates the performance. Figure 7 shows the BCR curve of the first 50 frames of Bus sequence.

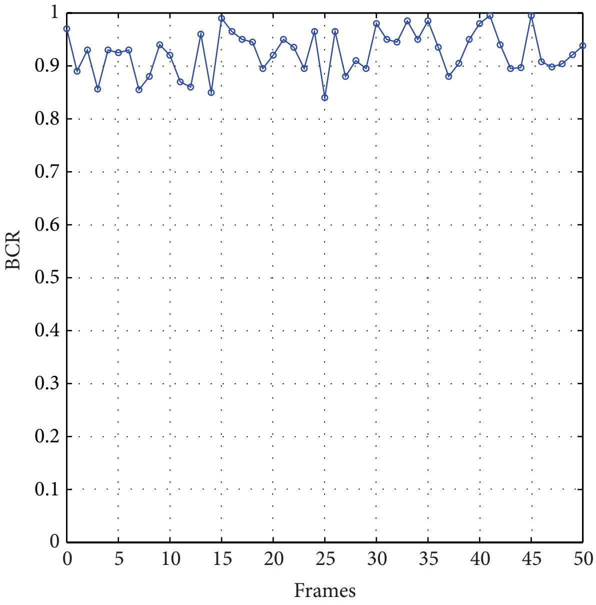


Figure 7: The BCR curve of Bus sequence.

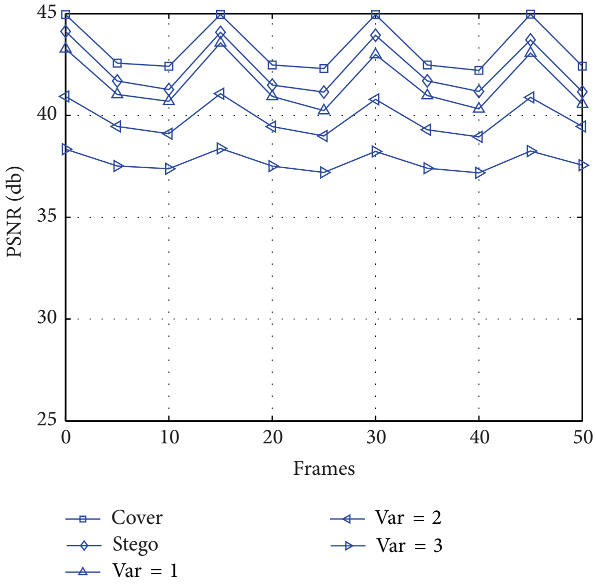


Figure 8: The PSNR curve of sequence with different noise level.

3.2.2. The Robustness against Noise. Video sequences are distorted by transmission noise. Hence the robustness against noise is a factor of one good video watermarking algorithm. In this experiment, Gaussian noises with different intensity are added into the luminance component to test the performance.

Suppose noise x is Gaussian distribution, namely, $x \sim G(\mu_X, \sigma_X^2)$; let $\mu_X = 0$; then adjust the σ_X^2 to test the robustness under different noise intensities.

The first 50 frames of Bus sequence are used as samples. The PSNR of sequence with different noises are shown in Figure 8, where σ_X^2 is 1, 4, and 9. It is obvious that the larger σ_X^2 is, the less PSNR is.

Figure 9 shows the BCR curve of video with different noise levels. P frames always have much lower BCR than I frames. This phenomenon is caused by the high compression efficiency of P frames. Thus the watermark in I frames has much robust than that in P frames (GOP includes 15 frames).

This algorithm embeds the same watermark in all frames in one scene. Thus even if the watermark is not detected in the P frames, then it can be detected in the I frames.

3.2.3. The Robustness against Temporal-Desynchronized Attack. This attack includes inserting or deleting and averaging and regrouping frames. Due to its easy manipulation and implementation, it is a common used attack in video

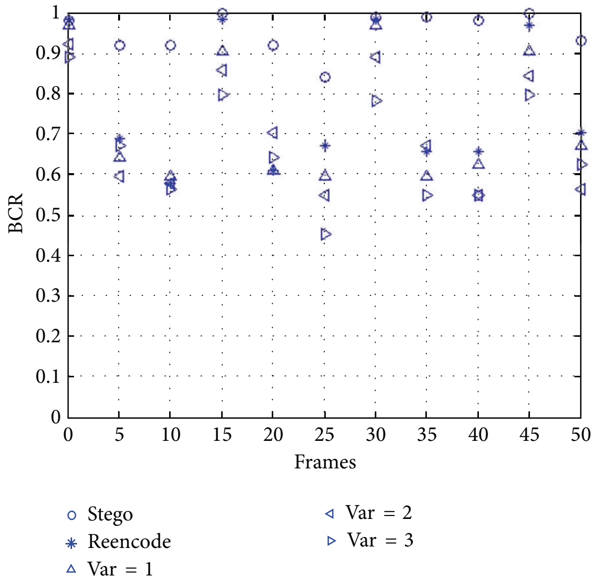


Figure 9: The BCR value with different noise level.

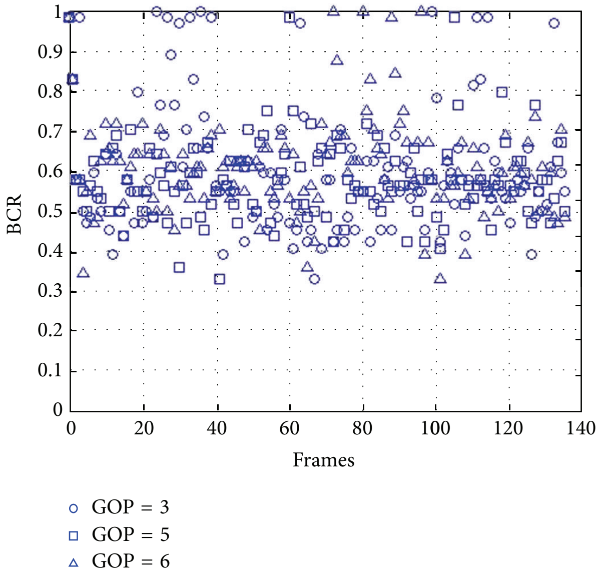


Figure 10: The BCR after deleting some frames.

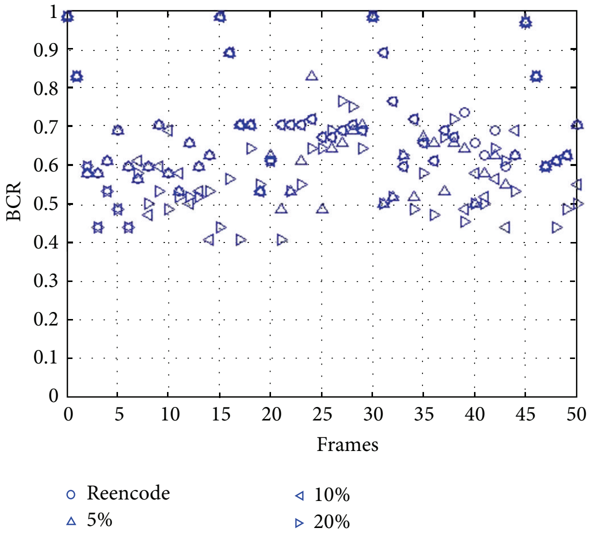


Figure 11: The BCR with frame regrouping.

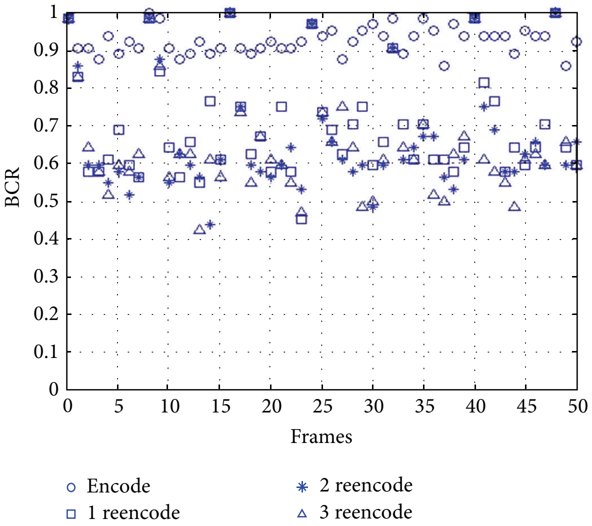


Figure 12: The BCR with recompression.

watermark. This attack can lead to the time-desynchronized attack; hence the watermark cannot be detected. Many video watermark algorithms design some sophisticated mechanism to resist this attack where the segmentation of scenes is the most valid measure to resist this type of attack.

Inserting or deleting frames may change the type of coding frame. It is well know that *P* frames use the prediction to improve the compression efficiency; thus the detection on *P* frames will be weaker than that on *I* frames. If *I* frames are not changed, then the watermark can be detected with high probability. Under different numbers of GOP frames, the BCR is shown in Figure 10 after randomly deleting some frames in Bus sequence. From that figure, when the GOP includes 3, 5, and 8 frames, the detected frames (BCR > 0.9) are 12, 4, and 3, respectively.

From this experiment, the proposed algorithm has the capability to resist the attack conducted by deleting frames.

Figure 11 shows the BCR value where some frames are regrouped with different percentages: 5%, 10%, and 20%. From this figure, frame regrouping does not change the position of most *I* frames; thus *I* frames and the following *P* frames keep the BCR unchanged.

3.2.4. The Robustness against Reencoding. Figure 12 shows the BCR with one, two, and three recompression attacks. From this figure, *I* frames have the highest stability.

3.3. Real-Time. Sometimes real-time plays an important role in applications. Of course, different applications have

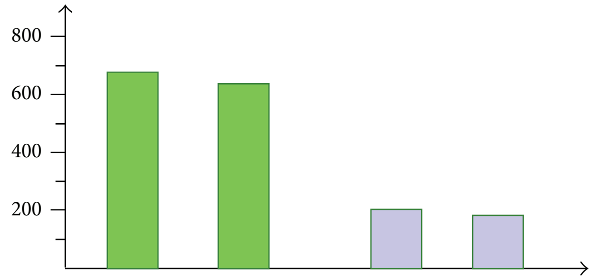


Figure 13: The time of video compression and decompression with and without watermark embedding (from right to left: the time of compression with and without watermark and decompression with and without watermark).

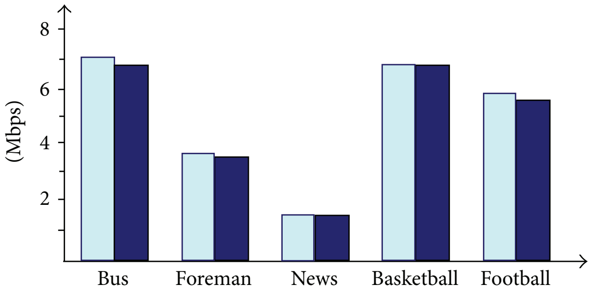


Figure 14: The bit-rate of different sequences with and without watermark.

different requirements. In this experiment, the mixed video toto is used. It includes 3979 frames. The time-complexity with and without watermark embedding is considered. The experimental environment is P4 2.66G 512 M memory. The results are shown in Figure 13. The watermark algorithm has only a small effect on the time-complexity. The decode speed achieves 22.3 frames per second without any code optimization.

3.4. Bit-Rate. Watermarking in video sequences often causes the increasing of the file size; thus how to control the increased amount is a common considered problem in video watermarking. This algorithm embeds the watermark bit by substituting those Huffman codes with less *bit1*. Moreover, the Huffman code with less *bit1* makes decreasing the total bits more probable. Hence, from the probability, the algorithm does not increase the bit-rate. Five sequences including Bus, foreman, news, basketball, and football are used as test sequences. The results are shown in Figure 14. The light bar indicates the original bit-rate without watermark embedding, and dark bar indicates the bit-rate with watermark embedding. Moreover, the bit-rate has a slight decrease. This agrees with the earlier analysis.

4. Conclusions

This paper proposes a real-time video watermarking algorithm based on scene segmentation. The experiments indicate that the proposed algorithm not only keeps the high quality of watermarked video but also provides strong robustness against recompression, noise, and time-desynchronized attacks. At the same time, the time-complexity on coding side is not larger than 10% and that on decoding side is not larger than 2%. Moreover, the algorithm does not increase the bit-rate of compressed video. This algorithm can be used in any video-related application. For example, it can be used in video surveillance to prevent somebody from forging surveillance video in all distributed sensor networks.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported in part by a scholarship from the China Scholarship Council of the Republic of China (file no. 201203070360), the Natural Science Foundation of China (no. 60803147), and the Major State Basic Research Development Program of China (no. 2015CB351804).

References

[1] X. Cheng and J. Liu, “Nettube: exploring social networks for peer-to-peer short video sharing,” in *Proceedings of the Conference on Computer Communications (INFOCOM ’09)*, pp. 1152–1160, IEEE, 2009.

[2] K. Xu, H. Li, J. Liu, W. Zhu, and W. Wang, “PPVA: a universal and transparent peer-to-peer accelerator for interactive online video sharing,” in *Proceedings of the 18th International Workshop on Quality of Service (IWQoS ’10)*, pp. 1–9, June 2010.

[3] X. Cheng, J. Liu, and C. Dale, “Understanding the characteristics of internet short video sharing: a youtube-based measurement study,” *IEEE Transactions on Multimedia*, vol. 15, no. 5, pp. 1184–1194, 2013.

[4] Y. Liu, K. P. Gummadi, B. Krishnamurthy, and A. Mislove, “Analyzing facebook privacy settings: user expectations vs. reality,” in *Proceedings of the ACM SIGCOMM Internet Measurement Conference (IMC ’11)*, pp. 61–70, ACM, Berlin, Germany, November 2011.

[5] D. Rotman and J. Preece, “The ‘WeTube’ in youTube—creating an online community through video sharing,” *International Journal of Web Based Communities*, vol. 6, no. 3, pp. 317–333, 2010.

[6] C. Gan and W. Wang, “Weibo or weixin? Gratifications for using different social media,” in *Digital Services and Information Intelligence*, vol. 445 of *IFIP Advances in Information and Communication Technology*, pp. 14–22, Springer, Berlin, Germany, 2014.

[7] M. Asikuzzaman, M. J. Alam, A. J. Lambert, and M. R. Pickering, “Imperceptible and robust blind video watermarking using chrominance embedding: a set of approaches in the DT CWT domain,” *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 9, pp. 1502–1517, 2014.

[8] M. Fallahpour, S. Shirmohammadi, M. Semsarzadeh, and J. Zhao, “Tampering detection in compressed digital video using watermarking,” *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 5, pp. 1057–1072, 2014.

[9] Chris, "Using video surveillance as evidence in court," *SECURITYBROS*, 2014, http://securitybros.com/using-video-surveillance-as-evidence-in-court/.

[10] Q. Wang, W. Zeng, and J. Tian, "A compressive sensing based secure watermark detection and privacy preserving storage framework," *IEEE Transactions on Image Processing*, vol. 23, no. 3, pp. 1317–1328, 2014.

[11] J. Cichowski and A. Czyzewski, "Reversible video stream anonymization for video surveillance systems based on pixels relocation and watermarking," in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV Workshops '11)*, pp. 1971–1977, Barcelona, Spain, November 2011.

[12] T. Bianchi and A. Piva, "Secure watermarking for multimedia content protection: a review of its benefits and open issues," *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 87–96, 2013.

[13] G. Doërr and J.-L. Dugelay, "A guide tour of video watermarking," *Signal Processing: Image Communication*, vol. 18, no. 4, pp. 263–282, 2003.

[14] M. D. Swanson, B. Zhu, and A. H. Tewfik, "Multiresolution scene-based video watermarking using perceptual models," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 4, pp. 540–550, 1998.

[15] X. Niu, S. Sun, and W. Xiang, "Multiresolution watermarking for video based on gray-level digital watermark," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 2, pp. 375–384, 2000.

[16] F. Hartung and B. Girod, "Watermarking of uncompressed and compressed video," *Signal Processing*, vol. 66, no. 3, pp. 283–301, 1998.

[17] G. C. Langelaar and R. L. Lagendijk, "Optimal differential energy watermarking of DCT encoded images and video," *IEEE Transactions on Image Processing*, vol. 10, no. 1, pp. 148–158, 2001.

[18] G. C. Langelaar, I. Setyawan, and R. L. Lagendijk, "Watermarking digital image and video data," *IEEE Signal Processing Magazine*, vol. 17, no. 5, pp. 20–46, 2000.

[19] C.-S. Lu, J.-R. Chen, H.-Y. M. Liao, and K.-C. Fan, "Real-time MPEG2 video watermarking in the VLC domain," in *Proceedings of the 16th International Conference on Pattern Recognition Proceeding*, vol. 2, pp. 552–555, 2002.

[20] D. Ye, C. Zou, Y. Dai, and Z. Wang, "A new adaptive watermarking for real-time MPEG videos," *Applied Mathematics and Computation*, vol. 185, no. 2, pp. 907–918, 2007.

[21] C.-S. Lu, J.-R. Chen, and K.-C. Fan, "Real-time frame-dependent video watermarking in VLC domain," *Signal Processing: Image Communication*, vol. 20, no. 7, pp. 624–642, 2005.

[22] S. D. Roy, X. Li, Y. Shoshan, A. Fish, and O. Yadid-Pecht, "Hardware implementation of a digital watermarking system for video authentication," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 2, pp. 289–301, 2013.

[23] H.-S. Jung, Y.-Y. Lee, and S. U. Lee, "RST-resilient video watermarking using scene-based feature extraction," *EURASIP Journal on Applied Signal Processing*, vol. 2004, Article ID 358092, pp. 2113–2131, 2004.

[24] B.-L. Yeo and B. Liu, "Rapid scene analysis on compressed video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, no. 6, pp. 533–544, 1995.

[25] V. T. Chasanis, A. C. Likas, and N. P. Galatsanos, "Scene detection in videos using shot clustering and sequence alignment," *IEEE Transactions on Multimedia*, vol. 11, no. 1, pp. 89–100, 2009.

[26] S. H. Liu, H. X. Yao, S. P. Zhang, and W. Gao, "Progressive quality hiding strategy based on equivalence partitions of hiding units," in *Transactions on Data Hiding and Multimedia Security VI*, vol. 6730 of *Lecture Notes in Computer Science*, pp. 58–82, Springer, Berlin, Germany, 2011.

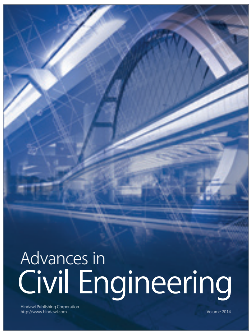
Journal of
Engineering

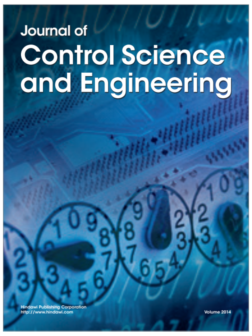
The Scientific
World Journal

International Journal of
Rotating
Machinery

Journal of
Sensors

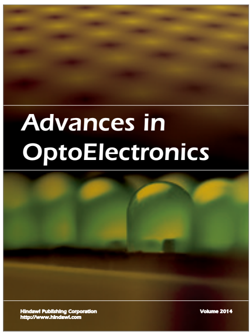
International Journal of
Distributed
Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

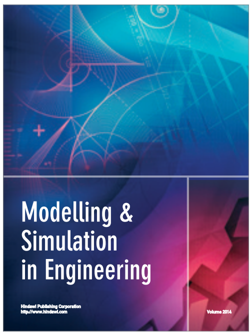
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

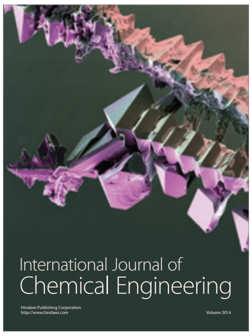

Advances in
OptoElectronics

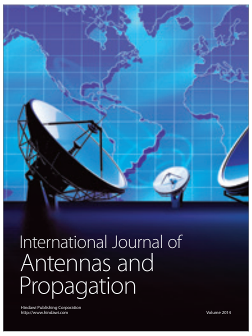
VLSI Design

International Journal of
Navigation and
Observation

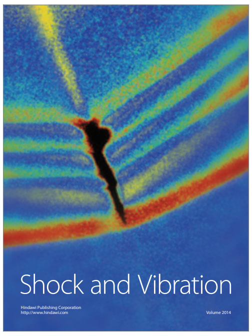

Modelling &
Simulation
in Engineering

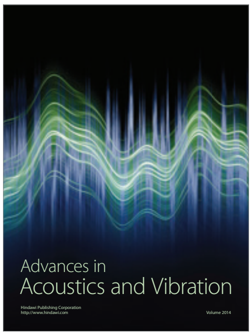
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration