# Mapping Energy Video Watermarking Algorithm Based on Compressed Domain

Lijun Wang, HongXun Yao, ShaoHui Liu, and Wen Gao

School of Computer Science and Technology,
Harbin Institute of Technology, Harbin 150001, P.R. China
{ljwang,yhx,shaohl}@vilab.hit.edu.cn, wgao@ict.ac.cn

**Abstract.** The paper presents a novel video watermarking scheme called Mapping Energy video Watermarking (MEW) for copyright protection. The watermark is embedded in the compressed domain and extracted directly from the bitstream without original video. In the proposed scheme, we select a part of Integer DCT quantized coefficients to construct the embedded space and embed the watermark into it by using Energy Mapping Function (EMF). During the process of embedding, resynchronization strategy and watermarking coding are used to guarantee the robustness of watermarking. The experimental results indicate that the scheme has strong robustness to the attacks such as re-encoding, frame dropping, frame rate changing and sharpening. The influence of the coding efficiency is almost unnoticeable. Besides, high watermark payload and low time complexity are advantages of the scheme.

## 1 Introduction

With the development of computer network, copyright protection of digital products like video and audio has become a hot research topic. Video watermarking is an efficient technology to protect the copyrights of digital video. Many watermarking schemes based on spatial domain [1–2] and frequency domain [3–4] have been developed and can be used both in image and video. Compared with image watermark, video watermarking schemes should have their own characteristics such as the robustness to compression and the frame operations (dropping, re-encoding, shifting etc). Video is always disseminated as a compressed format, so a video watermarking scheme performed in the compressed domain is necessary. Hartung [5] proposed a typical scheme in the compressed domain. They arranged a spread-spectrum watermark with the same size as one frame and divided it into 8*8 blocks. Each block is transformed by DCT and added into the corresponding video DCT block. Cross [6] proposed the watermark was embedded in VLC of I-frame with the proper payload. Besides, the watermarking schemes performed in the compressed domain include those embedded in the residual of motion vectors [7–8] and in the facial parameters of MPEG-4 bitsteam [9]. These video watermarking schemes can gain low computational complexity or improve the robustness to the compression, but they are not resistant to such attacks as re-encoding and frame dropping. Lagendijk [10]

K. Aizawa, Y. Nakamura, and S. Satoh (Eds.): PCM 2004, LNCS 3333, pp. 433–440, 2004.

has developed an algorithm called Extended Differential Energy Watermarking (XDEW), in which the watermark was embedded in both I-frames and P-frames. The algorithm is performed in the low bit-rate environment and has good performance on the robustness to re-encoding. But it is complicated from a computational standpoint. This paper proposes a novel video watermarking called adaptive Mapping Energy Watermarking (MEW) performed in the standard H.264 (JM 6.2) bitstream. We define the media data space is constructed by the quantized Integer DCT coefficients and the embedded space is a part of it. Every coefficient in the embedded space corresponds to one watermark bit and the value of the coefficient represents the energy of the watermark bit. This character of watermarking scheme gains it good performance on coding efficiency and watermark payload. The rest of this paper is organized as follows. In Section 2, MEW algorithm is explained in detail. In Section 3, the robustness of watermarking is analyzed and the experimental results are presented. Finally we present the conclusion of our experiments in Section 5.

## 2 MEW Algorithm

Video is a kind of media with a large data space and exits as a compressed format, so a video watermarking scheme in the compression domain is necessary. Figure 1 shows the diagram of our watermarking system based on the video encoder. From Figure 1, watermark and raw video are input as the original information and the watermarked compressed video and key (the quantized step is used to generate the key which guarantees the robustness to re-encoding at different bit rate) are output after embedding process; original watermark, watermarked video and the key are required during extracting process and the extracted watermark and detection value are output after extracting process.

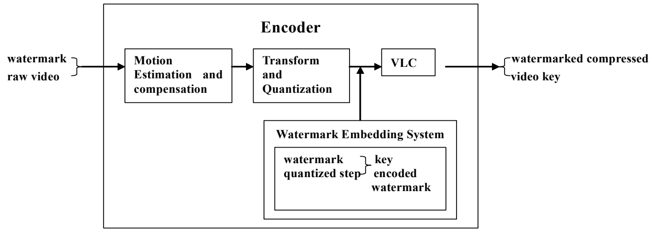


**Fig. 1.** The Diagram of Watermarking System

### 2.1 The Embedding Process

The embedding procedure is summarized as follows:

1. For a luma block, $M$Integer DCT quantized AC coefficients before a cut off point (cut_off) in zig_zag scanning order in a fixed scope ($a \leq abs(x) \leq b$) to construct the embedded space. If there are not as many as $M$ coefficients that can be selected, this block would not be embedded.

2. The watermark can be any useful information, and in our experiments it is a random sequence of 1s and −1s generated by a key. Corresponding to $M$selected coefficients, the watermark sequence is divided into several Groups of Watermark bits (GOW) with the size of $M$. Every selected coefficient corresponds to one watermark bit and we use Energy Mapping Function (formula 1) to embed the watermark.

$$f(x) = \begin{cases} x & a \leq abs(x) \leq c \quad , w = 1 \\ -c & c+1 \leq x \leq b \quad , w = 1 \\ c & -b \leq x \leq -(c+1) \; , w = 1 \\ x & c+1 \leq abs(x) \leq b \; , w = -1 \\ c & -c \leq x \leq -a \quad , w = -1 \\ c & c \leq x \leq b \quad , w = -1 \end{cases} \tag{1}$$

Where $x$ is the value of one coefficient, $w$ is one watermark bit, $a$ and $b$ are the bounds of embedded space and $c$ is the partition point (in our experiments, $a$, $b$ and $c$ are 2,3 and 2).

3. The same GOW is embedded in the 4 luma blocks of one MB to improve the robustness of watermarking if the blocks can be embedded.

## 2.2 The Extracting Process

In contrast with the embedding process, we use the following procedure to extract watermark bits.

1. In one luma block, $M$ coefficients ($a \leq abs(x) \leq b$) before a cut off point are selected. If there are not as many as $M$ coefficients, the block isn't watermarked.

2. The energy of one watermark bit is calculated by adding all the corresponding de-quantized values that represent it as the formula 2 and 3.

3. The watermark bit can be obtained according to the sign of the energy as formula(4).

$$E_i = \sum (Esign(abs(de\_quant(x)))) \tag{2}$$

$$Esign = \begin{cases} -, & c+1 \leq abs(x) \leq b \\ +, & a \leq abs(x) \leq c \end{cases} \tag{3}$$

$$Esign = \begin{cases} 1, & E_k > 0 \\ -1, & E_k \leq 0 \end{cases} \tag{4}$$

In the formula 2, $de\_quant(x)$ represents the de-quantized value of coefficient x, $Esign(x)$ represents the sign of $x$ according to its absolute value, $E_k$ represents the energy value corresponding to the Kth watermark bit and $W_k$represents the Kth watermark bit.

As the formulas show, the embedding energy is enlarged to guarantee extracting watermark bits exactly because the sign of energy will retain unmodified even though some of coefficients value exceed its original scope during re-encoding.

# 3 Experimental Results

We test the performance of MEW in terms of watermark capacity, robustness and visual quality. We use H.264 bitstream coded at 30 fps, and the spatial resolution of the video sequences in our experiments is $176 \times 144$ pixels (QCIF). The watermark used in the experiments is a random sequence of 1s or −1s generated by a key.

## 3.1 Watermark Payload and Visual Quality

We compare the video quality with and without watermarking as Table 1. 1. Asnry, Asnru and Asnrv are the average of PSNR of three chroma components, which weigh the visual quality of the compressed video. 2. Embedded_bits are the amount of watermark bits embedded in the video, which weighs the watermark payload. 3. BIR is Bit Increased Rate that weighs the increased amount after watermarked.

**Table 1.** The experiments results of 4 standard test sequences 300 $frames$, 30$fps$, $cut\_off = 35, M = 8$

| sequence / parameters | Foreman_qcif | | News_qcif | | Silent_qcif | | Akiyo_cif | |
|---|---|---|---|---|---|---|---|---|
| | original | Wm(1:1) | Original | Wm(1:1) | Original | Wm(1:1) | Original | Wm(1:1) |
| Asnry(db) | 38.79 | 38.77 | 40.19 | 40.14 | 39.30 | 39.24 | 41.72 | 41.71 |
| Asnru(db) | 42.23 | 42.23 | 42.99 | 42.97 | 42.15 | 42.13 | 43.54 | 43.51 |
| Asnrv(db) | 43.95 | 43.94 | 43.62 | 43.63 | 42.97 | 42.93 | 44.56 | 44.58 |
| Totalbits(bits) | 3261856 | 3277552 | 1751176 | 1764440 | 1833064 | 1841386 | 726968 | 736928 |
| Bit rate(kbps) | 328.37 | 329.95 | 176.29 | 177.63 | 184.54 | 185.37 | 73.18 | 74.19 |
| Total time(s) | 218.408 | 227.689 | 193.778 | 209.255 | 208.983 | 207.750 | 191.128 | 198.176 |
| Embedded_bits | | 2928 | | 3176 | | 2296 | | 1736 |
| BIR (%) | | 0.48116 | | 0.76011 | | 4.476 | | 1.380 |
| WBR(bps) | | 294.76 | | 319.73 | | 231.1 | | 174.77 |

$$BIR = \frac{watermarked\_BR - original\_BR}{original\_BR} \times 100\% \quad (5)$$

Where $watermarked_BR$denotes the Bit Rate with watermark and $original_BR$denotes the Bit Rate without watermark. This parameter weighs the effect of compression efficiency caused by watermarking.

4.WBR denotes the Watermark Bit Rate, which weighs the watermark payload.

$$WBR = \frac{watermarked\_BR \times Embedded\_Bits}{total\_bits} \times 100\% \quad (6)$$

Where $total\_bits$ is the bits of watermarked video.

From Table 1, the coding efficiency of H.264 is very high and the degradation caused by watermarking is almost unnoticeable (the modification of PSNR and BIR are very small compared with encoding without watermarking). BIR is limited in 0.05 percent and WBR can get the largest payload of 300 bps (news, BR=177.63kbps). WBR can be large if the parameters are set properly.

Compared with XDEW algorithm, MEW has better visual quality and higher watermark payload. The following figures show PSNR and BIR curves. In fact, there is no law from the curves and the effect of PSNR and BIR is different with different video. But as a whole, PSNR is high and BIR is low.

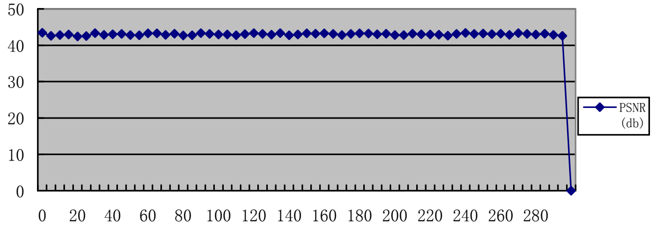


**Fig. 2.** PSNR of watermarked "Claire.yuv" video

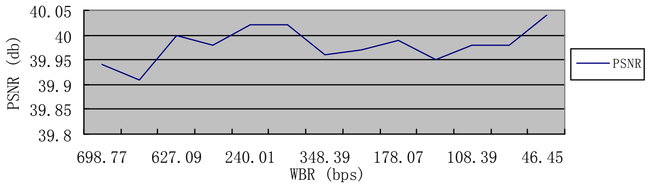


**Fig. 3.** PSNR at different WBR *News.yuv*

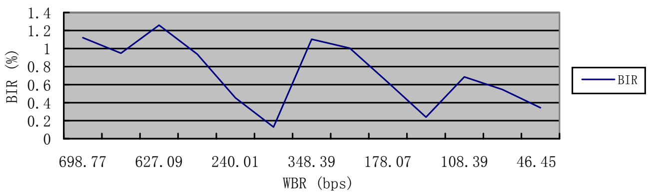


**Fig. 4.** BIR at different WBR *foreman.yuv*

### 3.2 The Robustness of Watermarking

As the experimental results prove, some non-watermarked MBs are included and some embedded MBs are discarded during embedding when the watermarked video is attacked by sharpening, frame dropping, re-encoding etc. When this scenario occurs, the synchronization between embedding and extracting is destroyed. We use re-synchronization strategy and watermark coding to guarantee the synchronization.

### 3.2.1 Re-synchronization Strategy

Re-synchronization strategy includes re-embedding and optimal retrieving strategies.

1. Re-embedding strategy:

As the description of embedding procedure, we embed a GOW in a luma block. For the continuous $P$ MBs, we embed the same GOW in every luma block of them and these $P$ MBs construct a Group of Embedded MBs (GEMB). When one MB is destroyed, GOW still can be extracted from other MBs in the same GEMB.

2. Optimal retrieving strategy:

We define a parameter as Dependable GOW ($Dep\_GOW$) initiated by the correct GOW. For every Current extracted GOW ($Curr\_GOW$), the Similar Rate between $Dep\_GOW$ and $Curr\_GOW$ is calculated as formula 7. If SR is no less than T, the current MB with the extracted GOW belongs to the current GEMB and $Dep\_GOW$ is updated as formula 8; otherwise, the following Skipped Number (Max_SN) GOWs are extracted and compared each other, then the most similar two of Max_SN GOWs are found and the corresponding MBs belong to the next GEMB.

$$SR = (Dissimilar\_num) + Len(wm) = \begin{cases} \geq T, & , Same \\ < T, & , Different \end{cases} \tag{7}$$

Where $Dissimilar\_num$ denotes the number of different bits in the two GOWs and $len(wm)$denotes the length of one GOW ($M$). In our experiments, we set T as 0.75 to obtain good performance.

$$Dep\_GOW = Dep\_GOW + Curr\_GOW \tag{8}$$

Where $Dep\_GOW$ denotes the dependable GOW, $Curr\_GOW$ denotes the current extracted GOW.

From the process described above, SR is the criterion of segmenting GEMBs.

### 3.2.2 Watermark Coding

When the watermarked video is attacked, the embedded space will be changed. We encode the watermark with Reed-Solomon coding due to its strong ability of correction. In our experiments, we set parameters of Reed-Solomon coding to correct at most 2 bits when 3 original bits are coded. Strong robustness can be obtained by sacrificing high watermark payload since the coded watermark bitstream is much larger than original watermark bitstream.

### 3.2.3 Experimental Results

In table 2, ABER denotes the Bit Error Rate after re-encoding which weighs the robustness of re-encoding (formula 9).

$$ABER = \frac{Error\_bits}{Embedded\_bits} \times 100\% \tag{9}$$

Where $Error\_bits$ denotes the number of error extracted watermark bits.

**Table 2.** The experimental results with R-S code for the 4 standard test video sequences at the fixed parameters $Max_S N = 2, = 4(176 * 144$ pixels, 30fps, 66 frames)

| sequence / parameters | Foreman_qcif | | News_qcif | | Silent_qcif | | Akiyo_cif | |
|---|---|---|---|---|---|---|---|---|
| | original | Wm(1:1) | Original | Wm(1:1) | Original | Wm(1:1) | Original | Wm(1:1) |
| Asnry(db) | 38.76 | 38.72 | 40.11 | 40.04 | 39.28 | 39.22 | 41.60 | 41.55 |
| Asnru(db) | 42.10 | 42.08 | 42.60 | 42.60 | 42.01 | 42.02 | 43.39 | 43.39 |
| Asnrv(db) | 43.85 | 43.86 | 43.41 | 43.41 | 42.93 | 42.91 | 44.46 | 44.44 |
| Totalbits(bits) | 596504 | 597880 | 356704 | 358336 | 400488 | 403536 | 174592 | 176336 |
| Bit rate(kbps) | 279.61 | 280.26 | 167.21 | 167.97 | 187.71 | 189.16 | 81.84 | 82.66 |
| Total time(s) | 181.312 | 185.579 | 164.456 | 179.369 | 165.766 | 175.925 | 154.373 | 176.854 |
| Embedded_bits | | 12 | | 15 | | 12 | | 2 |
| BIR (%) | | 0.232 | | 0.4545 | | 0.7724 | | 1.002 |
| ABER (%) | | 46.67 | | 51.136 | | 35.577 | | 62.5 |
| ABER with RS coding ( %) | | 0 | | 0 | | 0 | | 0 |
| WBR(bps) | | 5.625 | | 7.0312 | | 5.625 | | 2.16 |

From Table 2, the effect of the performance of encoding efficiency caused by watermarking is very small compared with original encoding efficiency and the robustness to re-encoding with RS coding is very strong (ABER without RS coding is much larger than that with RS coding). From the point, Reed-Solomon code is very optimal to correct most error bits and it can gain the optimal performance if P is selected properly.

When the watermarked video is attacked, the synchronization between the embedding and extracting will be destroyed. Moreover, the effect of synchronization is different with different attacks.

Figure 5 shows BER curves at the attacks of re-encoding, frame rate changing, frame dropping and sharpening (original video is compressed at the quantized step of 27; re27 means the video is re-compressed at 27 and re10 means it is re-compressed at 10; frame dropping and sharpening mean the video is decompressed first, then an arbitrary frame is dropped and all the I frames are sharpened, finally re-compressed at 27).

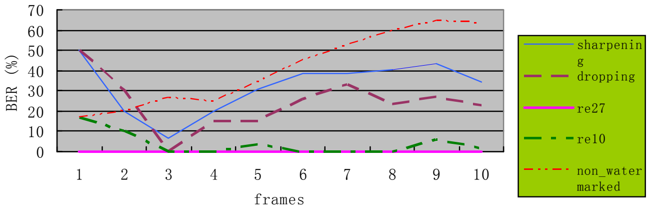


**Fig. 5.** BER at the attacks of sharpening, frame dropping and re-encoding *News.yuv*

Where non_watermark curve represents the BER of video without watermark. From the figure 5, BER is limited in 20 when the video is re-compressed at quantized step of 27 and 10. BER is oppositely large at first frames in dropping and sharpening curves, but it is decreased when I frames are increased.

## 4 Conclusion

This paper presents a novel blind video watermarking algorithm MEW. MEW algorithm includes re-embedding, optimal retrieving and watermark coding. These strategies are used to guarantee the encoding efficiency and the robustness of watermarking. As the experimental results prove, the performance of MEW is very excellent. MEW algorithm has such advantages as low time complexity, large watermark payload and strong robustness to re-encoding. We only use MEW to embed the watermark in I frames, in fact, P or B frames can also be used to embed more watermark bits or improve the robustness of watermarking. We use H.264 compression standard to test our algorithm and it can be extended to other standards as MPEG-2, MPEG-4 etc.

## References


1. R.van Schyndel, A. Tirkel and C. Osborne: A digital watermark. IEEE ICIP, Vol.2, (1994) 86–90
2. B.C. Mobasseri, M. J. Sieffert and R. J. Simard: Content authentication and tamper detection in digital video. IEEE ICIP, Vol.1, (2000) 458–461
3. I.J.Cox, J.Killian, T.Leighton and T.Shamoon: Secure spread spectrum watermarking for multimedia. IEEE Trans. On Image Processing, Vol.6 (12), (1997) 1673–1687
4. Xiaoyun Wu, Wenwu Zhu, Zixiang Xiong and Ya-Qin Zhang: Object-based multiresolution watermarking of images and video. IEEE ISCS 2000, Vol.1, (2000) 212–215
5. F. Hartung and B. Girod: Digital watermarking of MPEG-2 coded video in the bitstream domain. ICASSP-97, Vol.4, (1997) 2621–2624
6. D. Cross and B. G. Mobasseri: watermarking for sef-authentication of compressed video. IEEE ICIP, Vol.2, (2002) 913–916
7. Zhuo zhao, Nenghai Yu and Xuelong Li: A novel video watermarking scheme in compressed domain based on fast motion estimation. IEEE ICCT, (2003)1878–1882
8. Yuanjun Dai, Lihe Zhang and Yixian Yang: A new method of MPEG video watermarking technology. IEEE ICCT, (2003)1845–1847
9. F. Hartung, P. Eisert and B. Girod: Digital watermarking of MPEG-4 facial animation parameters. Computers and Graphics, Vol. 22, (1998) 425–435
10. I. Setyawan and R. L. Lagendijk: Low bit rate video watermarking using temporally extended differential energy watermarking(DEW) algorithm. Proc. Security and Watermarking of Multimedia Contents III, Vol. 4314, (2001) 73–84