Key-based Video Watermarking System on MPEG-2

Shiau-Jen Weng, Ta-Te Lu, and Pao-Chi Chang
Department of Electrical Engineering, National Central University, Taiwan

ABSTRACT

In this paper, we propose a key-based video watermarking system in which the watermark embedding and the video encoding are processed at the same time on MPEG-2. Since the watermark information would propagate to inter-frames through the motion compensated coding, the watermark is embedded in a single intra-frame but can be extracted from all frames in the same group of pictures (GOP). The watermark is embedded in the low frequency DCT coefficients of the intra-frames based on the block polarity. The block polarity is Tri-state Exclusive-Or (TXOR) with the watermark to generate the secret key, which labels the block locations of the embedded watermark. In the decoding end, the block polarity over a GOP is calculated by a weighted voting procedure according to the frame weighting. Finally, the watermark over a GOP can be obtained by TXOR operation of the key and the block polarity. The simulation results show that the system has great imperceptibility that the PSNRs of the watermarked frames are almost the same as the un-watermarked ones and more accurate normalized correlation (NC) can be obtained as well.

Keywords: Robust watermarking, video watermarking, DCT, MPEG-2.

1. INTRODUCTION

Digital watermarking is one of the solutions to protect intellectual property of digital multimedia data. In the beginning, numerous watermarking algorithms were proposed for image, but not many for video [1][2]. Only recently, more and more methods for video watermarking were developed [3]-[9]. Although image watermarking is somewhat similar to video watermarking, there are certain different considerations in the design of watermarking approach. The major requirements of image watermarking for copyright protection are robustness, imperceptibility, and low false alarm rate [10]. In addition to these requirements, complexity and real-time detection are also necessary for video watermarking. Blind detection is a much more important issue for video watermarking than image watermarking [2]. The reason is that video usually contains such a large amount of data that it is impractical to store original video for watermark extraction. In addition, video sequences usually have high temporal correlation so that video compression standards, such as H.263 and MPEG-2, compress video by removing the temporal redundancy using the motion compensation coding. It is the temporal relationship between video frames that could be considered and utilized in video watermarking as well. In [3], Hsu and Wu proposed to embed watermarks into intra- and inter-frames with different residual masks to take advantage of prediction types of MPEG bitstreams.

When designing a watermark algorithm for MPEG-2, three locations could be chosen to place watermark embedding block [4]. Watermarks can be embedded in video sequences before or after video are sent into MPEG-2 encoder, or watermark can be embedded with the modified MPEG-2 encoder. Then the watermark can usually be extracted before, with, or after MPEG-2 decoder symmetrically. Each of these schemes has its advantages and drawbacks. The first scheme that watermark embedding/extraction is performed in raw videos is less constrained by considering video coding specification. Namely, watermark can be embedded in Discrete Fourier Transform (DFT) domain [5], spatial domain, etc., rather than Discrete Cosine Transform (DCT) domain only. For the second scheme, where watermark is embedded after MPEG-2 encoding, i.e., in bitstream directly, the computation complexity could be the lowest among these three schemes. However, embedding watermark with arbitrary methods may increase bitstream size. Moreover, the video quality is much prone to degrading due to the change of the motion vector or the reference frames that will be referred in the motion compensation, so Hartung and Girod [6] proposed a scheme that can avoid visible artifacts by adding a drift compensation signal. The watermark embedded in the bitstream can be in DCT coefficients [6], motion vectors [7][8], or in variable length codewords [9]. The third scheme embeds watermark into coefficients after DCT [4] or quantization. It takes time to partially decode compressed video for watermark embedding if the original video is

Security and Watermarking of Multimedia Contents V, Edward J. Delp III, Ping Wah Wong, Editors, Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 5020 (2003)
© 2003 SPIE-IS&T · 0277-786X/2003/$15.00

compressed ones but can avoid the bitrate constraint and the drift problem. Furthermore, watermark embedding and video compression could be realized at the same time. The proposed method here belongs to this kind of scheme.

In this paper, we propose a key-based video watermarking system on MPEG-2. We utilize the property of motion compensation coding that the information in intra-frame would propagate to the succeeding inter-frames in the same GOP. Therefore, watermark is embedded just in intra-frame and can be extracted from all frames. However, not all information of inter-frames can be used since the variation of the video sequences may be large. To consider this in watermark extraction, we apply the soft decision to determine the block polarity over a GOP based on the frame weighting. Consequently, more accurate watermark is extracted through this voting procedure.

The rest of this paper is organized as follows. In Section 2, the video watermarking system is described. The proposed watermark embedding and extraction algorithm are described in Section 3. Section 4 shows the simulation results, and Section 5 concludes this paper.

2. VIDEO WATERMARKING SYSTEM ON MPEG-2

MPEG-2 consists of three picture types [11]: Intra-coded picture or intra-frame (I frame), Predictive coded picture (P frame), and Bidirectionally predicted picture (B frame). I frame is encoded in a way similar to image compression. P frame is motion compensation coded by referring to the nearest previous I/P frame, while B frame refers to forward and backward I/P frames. This motion compensation coding can remove the temporal redundancy so that the amount of video data can be compressed efficiently. In the decoder, the temporal redundancy is added back to get the full frame. Because of the motion compensation, the key-based watermark that is embedded only in I frame will propagate to P/B frames (inter-frames).

In order to compress video sequences and embed watermarks at the same time, the key-based watermark-embedding algorithm is integrated in MPEG-2 [11] encoder, which is depicted in Fig. 1. In MPEG-2 encoder, intra-frames are discrete cosine transformed, quantized, and then variable length encoded, while inter-frames are first motion-estimated to get the motion vectors and the residual frames which are compressed by the same procedure as intra-frames. Since the inter-frames are predicted from the intra-frames for motion estimation, the watermark embedded in the intra-frames would propagate to the inter-frames. Hence, only the DCT coefficients of the intra-frames are sent to the watermark-embedding block, in which keys are obtained and stored as the side information that will be sent to the watermark extractor.

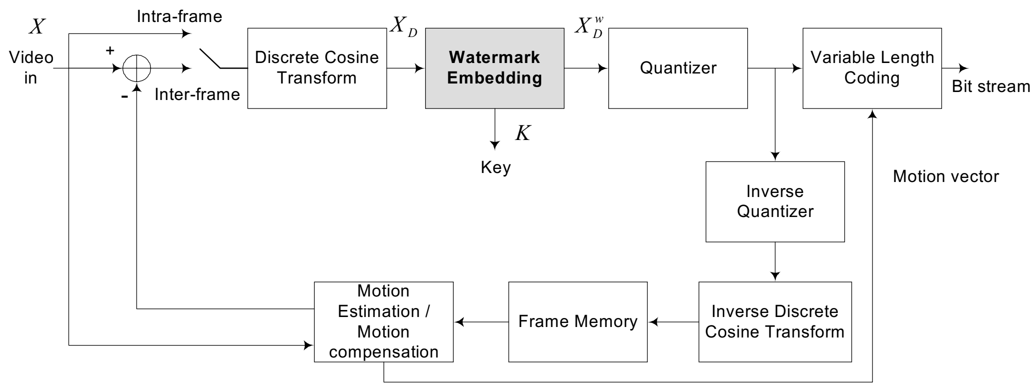

Fig. 1 MPEG-2 encoder with the watermark embedding

Watermarks are extracted after the bitstream are decoded with MPEG-2 decoder as shown in Fig. 2. According to the correspondent key per GOP and the decoded video data obtained in MPEG-2 decoder, the watermarks are extracted from intra- and inter-frames. The algorithm of this key-based watermarking embedding and extraction are described in detail in next section.

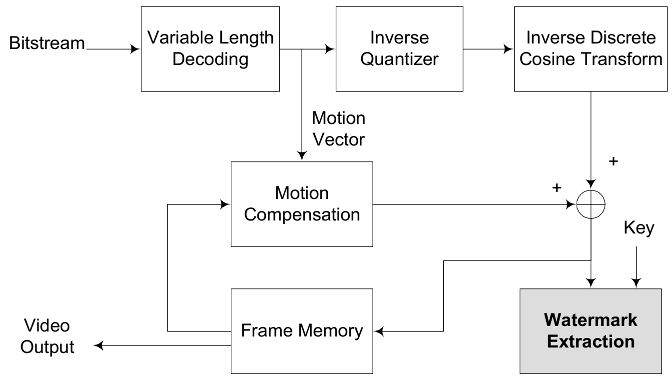

Fig. 2 MPEG-2 decoder with the watermark extraction

3. KEY-BASED WATERMARK EMBEDDING AND EXTRACTION

MPEG-2 uses 8x8 DCT and provides compressed video in high quality, so a key-based image watermarking algorithm operated in DCT domain [12] with good imperceptibility is proper to be applied here.

3.1 WATERMARK EMBEDDING

The block diagram of the watermark embedding procedure is illustrated in Fig. 3. Four main functions are included: Subblock composition for selecting proper coefficients that are capable of resisting attacks, such as filtering and compression; block polarity mapping to decide if the subblock is suitable to embed watermark; key generation for recording which subblock has been embedded watermark and thus providing the blind detection property; deadzone evacuation for enhancing the subblock polarity to achieve higher robustness against attacks. In addition, random permutation of watermark sequence and subblocks are performed to disperse the spatial relationship for higher security. The procedure of watermark embedding is detailed below.

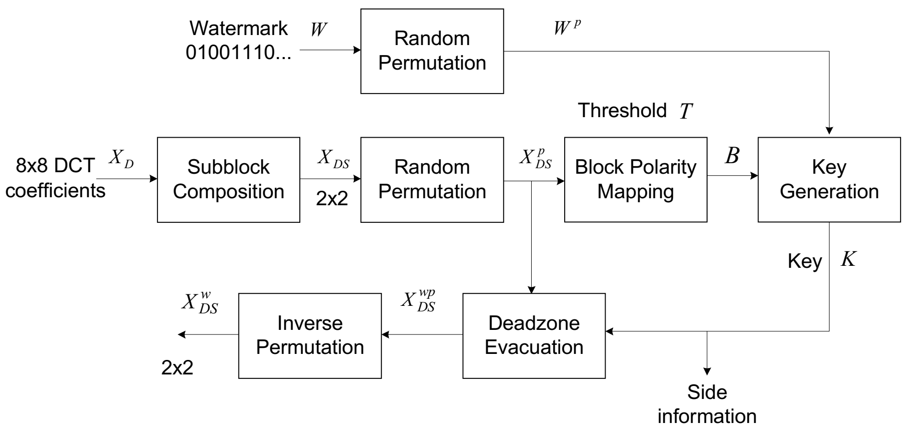

Fig. 3 The block diagram of a key-based watermark embedding procedure

The watermark logo used here is a $H_w \times L_w$ bi-level image, which can be represented as a one-dimension sequence $W = \{W_l, l = 1,2,...,H_w \times L_w\}$consisting of binary values "1" and "0", where l represents the binary sequence index. By random permutation, we get the watermark after permutation $W^p = \{W_l^p = permu(W_l), l = 1,2,...,H_w \times L_w\}$ from the watermark W, where $permu(\cdot)$ denotes the random permutation. For the video X consists of S pictures, each picture contains $H \times L$ pixels and thus comprises $N = \frac{H \times L}{8 \times 8}$ blocks, the DCT coefficients can be expressed as $X_D = \{dct_b^i(j) = dct_b^i(zigzag(x,y)), i = 1,2,...,S, b = 1,2,...,N, j = 1,2,...,64, x,y = 1,2,...,8\}$, where i represents the i-th frame, b represents the b-th block in that picture, j is the index of DCT coefficient, *zigzag(x,y)* is the zigzag order to scan the DCT coefficients from DC coefficient to the high frequency AC coefficients. The DCT coefficients X_D of the intra-frames, which are obtained in Section 2, are first sent to subblock composition.

3.1.1 Subblock Composition

In subblock composition, the four lowest frequency DCT coefficients are chosen to form subblocks because they are more capable of resisting attacks. The obtained subblocks can be expressed as $X_{DS} = \{C_b^i(j) = dct_b^i(j), i = 1,2,...,S, b = 1,2,...,N, j = 1,2,3,4\}$, where $C_b^i(j)$ is the selected coefficient, and b represents b-th block/subblock in that picture. Then the random permutation is applied to the subblocks X_{DS} to get X_{DS}^p, and X_{DS}^p performs the block polarity mapping.

3.1.2 Block Polarity Mapping

The first step in block polarity mapping is to calculate the threshold T, the mean of DCT coefficients in the same positions of every intra-frame, i.e., for $i = 1, 1+M, 1+2\times M, ..., 1+(m-1)\times M, ..., 1+\left(\left\lfloor \frac{S}{M} \right\rfloor - 1\right)\times M$, where m represesnts the m-th intra-frames/GOP and M is the total number of frames in one GOP, the four thresholds of the i-th frames are calculated as (1).

$$T_i(j) = \frac{1}{N}\sum_{b=1}^{N} C_b^i(j), j = 1,2,3,4 \tag{1}$$

Though the following procedures are performed through the whole intra-frames in video sequences, i is omitted for convenience from now on. For each subblock, the selected DCT coefficients $C_b(j)$ are compared to the threshold $T(j)$ to find out the coefficient polarity $P_b(j)$.

$$P_b(j) = sign(C_b(j) - T(j)) \tag{2}$$

By this definition, a DCT coefficient that is larger than the threshold is regarded as positive; a coefficient smaller than the threshold is negative. Thus the block polarity B_b, which denotes the b-th block polarity, can be acquired according to the mapping rules as follows: If all or almost all coefficients in one block are larger than the threshold, its polarity is defined as positive; if all or almost all coefficients in one block are smaller than the threshold, the polarity of this block is defined as negative; otherwise, the polarity of the block is undefined, and it infers that the block is not suitable for watermark embedding because its polarity is ambiguous that it could be changed easily. That is,

$$B_b = \begin{cases} 1 & ,if \sum_{j=1}^{4} P_b(j) \geq 3. \\ 0 & ,if \sum_{j=1}^{4} P_b(j) \leq -3. \\ U & ,\text{otherwise}. \end{cases} \tag{3}$$

Therefore, the block polarity $B = \{B_b, b = 1,2, ..., N\}$ is obtained and then used to generate the key.

3.1.3 Key Generation

A Tri-state Exclusive-Or (TXOR) operation, the truth table of which is listed in Table 1, is performed by the watermark W^p and the block polarity B to generate the key $K = \{K_g\}$ bit by bit:

$$K_g = \begin{cases} W_l^p \oplus B_b & ,B_b = 0,1 \\ 0 & ,B_b = U \end{cases} \tag{4}$$

where $\oplus$ denotes XOR. While the indexes g and b are both increased step by step, the index l of the watermark W^p is increased only when K_g has been assigned 1, i.e. the watermark is embedded. As a result of that, the key records the block positions of watermark embedded, so that the watermark can be extracted with the key K rather than the original video. The key K is stored and sent as the side information to the extraction end.

W^p	B_b	K_g
1	0	1
1	1	0
0	0	0
0	1	1
1	U	0
0	U	0

Table I The truth table of TXOR

3.1.4 Deadzone Evacuation

Both the block polarity B and the key K, obtained in the preceding two steps, are used to determine if the selected low frequency coefficients of the corresponding subblock are modified or not. The main purpose of coefficient modification is to move out coefficients from deadzone and thus enhance the robustness of the block polarity to against attacks. The coefficient modification can be categorized into two classes according to the block polarity B_b for the *b*-th block:

1) When $K_g = 1$ and $B_b = 1$, if $0 \le C_b(j) - T(j) \le t$, then $C_b(j)$ is modified to $T(j) + t$.

2) When $K_g = 1$ and $B_b = 0$, if $-t \le C_b(j) - T(j) \le 0$, then $C_b(j)$ is modified to $T(j) - t$.

$[T(j) - t, T(j) + t]$ is the defined deadzone. Finally, inverse permutation is performed on the subblocks, which are then placed back into 8x8 blocks to continue the following MPEG-2 encoding procedures mentioned in section 2.

3.2 WATERMARK EXTRACTION

The block diagram of watermark extraction is depicted in Fig. 4. First of all, the decoded video sequences are obtained in MPEG-2 decoder, and the key is received from the embedding end. Taking these data as input, watermark is extracted by six steps: Discrete cosine transform, subblock composition, random permutation, block polarity voting, watermark retrieval, and inverse permutation. The decoded video Y is discrete cosine transformed to get $Y_D = \{dct_b'^i(j) = dct_b'^i(zigzag(x,y)), i = 1,2, ..., S, b = 1,2, ..., N, j = 1,2, ..., 64, x,y = 1,2, ..., 8\}$. Next, the four lowest frequency DCT coefficients are selected from Y_D to form subblocks $Y_{DS} = \{C_b'^i(j) = dct_b'^i(j), i = 1,2, ..., S, b = 1,2, ..., N, j = 1,2,3,4\}$. Random permutation is performed on Y_{DS} to get

Y_{DS}^p. Actually, these two steps are the same as mentioned in Section 3.1. Afterwards, it is the block polarity voting that gets the block polarity in every GOP. The GOP-based block polarity and the correspondent key are TXOR operated to retrieve the watermark $W'^p = \{W_l'^p\}$: If $K_g = 1$ then it performs XOR with B_b' to obtain $W_l'^p$; otherwise, B_b' is erased. At last, inverse permutation is applied to the watermark W'^p to obtain the watermark W'. The voting procedure for block polarity, which is the most important step, collects information in one GOP to improve the accuracy of watermark extraction. The details are described in the following.

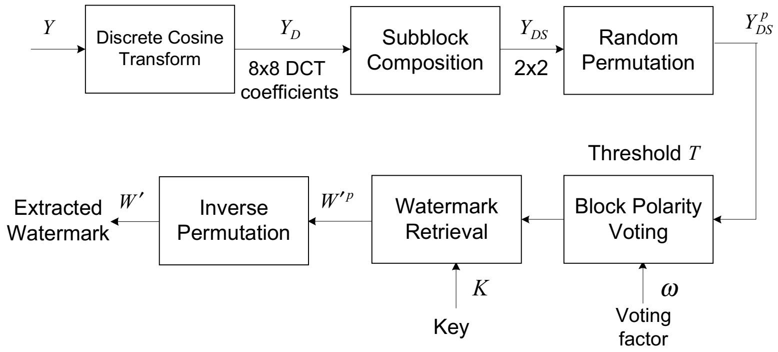

Fig. 4 The block diagram of a key-based watermark extraction procedure

The first step of block polarity voting is to calculate the threshold, which is the same as (1), i.e. computing the averages of the DCT coefficients in the same position,

$$T_i(j) = \frac{1}{N}\sum_{b=1}^{N} C_b'^i(j),\ j = 1,2,3,4,\ i = 1,2,\ldots S. \tag{5}$$

where *i* represents the *i*-th frame. However, in the extraction, not only intra-frame but also inter-frames will go through this step, namely threshold calculation. The block polarity is obtained from all frames in a GOP rather than only from intra-frame by the soft decision for the coefficient polarity $P_b(j)$.

3.2.1 Soft Decision

When watermarked video is attacked intentionally or unintentionally, the coefficient polarity $P_b(j)$ may be changed. That could result in wrong detection for watermark extraction. Thus, (2) is modified as the soft decision mode. The difference between DCT coefficients and the thresholds are accumulated with multiplying a voting factor ω, and then the accumulated result is averaged to determine the block polarity. For *M* frames in one GOP, the voted coefficient polarity $P_{bv}(j)$ is computed as

$$P_{bv}(j) = sign\left(\frac{1}{M}\sum_{gop}\omega\cdot\left(C_b^i(j) - T_i(j)\right)\right) \tag{6}$$

where *i* denotes the *i*-th frame. Then the block polarity B_b' can be acquired using the voted coefficient polarity $P_{bv}(j)$ as follows

$$B_b' = \begin{cases} 1 & ,if\ \sum_{j=1}^{4} P_{bv}(j) \ge 3. \\ 0 & ,if\ \sum_{j=1}^{4} P_{bv}(j) \le -3. \\ \mathrm{U} & ,\text{otherwise.} \end{cases} \tag{7}$$

Determining the voted coefficient polarity $P_{bv}(j)$ by the soft decision with proper voting factor ω, most of the errors will be cancelled out except there are serious errors. The voting factor ω can be designed from two points of view either in frame level or in block level. In this paper, we propose a simple model considering the frame weighting ω_f on the frame level.

3.2.2 Frame Weighting

It is assumed that the motion in each block of a video sequence has approximately constant speed, then both the picture type of frame and the temporal distance are used to determine the frame weighting $\omega_f(i)$. Because the watermark is embedded in I frame, the frame weighting of I frame is set as 1 while inter-frames get the lower frame weighting that defined as

$$\omega_f(i) = \begin{cases} \dfrac{a}{D_t(i)} & \text{, for P/B frames} \\ 1 & \text{, for I frames} \end{cases} \tag{8}$$

where *i* represents the *i*-th frame, *a* is a constant, and D_t is the function of the temporal distance. D_t is assigned to every frame in a GOP by P-B groups. For any P-B group shown as in Fig. 5, D_t is defined as the maximum temporal distance to the I frame.

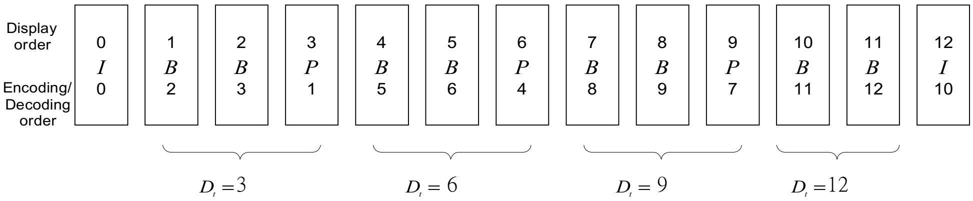

Fig. 5 The assignment of D_t according to the temporal distance

4. SIMULATION RESULTS

For simulation, we use six video sequences of CIF format: Akiyo, Hall monitor, Mother & daughter, Foreman, News and Stefan. The first three video sequences belong to Class A, and Foreman and News belong to Class B while Stefan is of Class C. Watermarks of size 16x16 are embedded in the luminance component (Y) at 2Mbps (bits/second).

PSNRs (dB)		Overall	I	P	B
Akiyo (Class A)	un-watermarked	47.465	47.173	48.132	47.251
	watermarked	47.215	45.773	47.996	47.109
Mother & daughter (Class A)	un-watermarked	44.683	45.127	45.497	44.313
	watermarked	44.618	44.682	45.473	44.285
Hall_monitor (Class A)	un-watermarked	40.020	40.255	40.993	39.618
	watermarked	40.009	40.164	40.987	39.616
Foreman (Class B)	un-watermarked	38.739	39.035	39.877	38.271
	watermarked	38.722	38.935	39.864	38.264
News (Class B)	un-watermarked	44.153	43.082	44.727	44.084
	watermarked	44.012	42.409	44.660	43.989
Stefan (Class C)	un-watermarked	31.791	33.088	33.087	31.134
	watermarked	31.786	33.077	33.080	31.130

Table II The average PSNRs comparison between watermarked and un-watermarked of overall/I/P/B frames at 2Mbps

Table II shows the PSNRs of watermarked and un-watermarked luminance components of six video sequences. Both are encoded and decoded to compare the PSNR before and after MPEG-2 compression. We can observe the effect of watermark embedding on perceptual quality. The overall PSNRs of watermarked and un-watermarked ones are quite similar and thus show the high imperceptibility of this watermark system. If PSNRs of I/P/B frames are traced separately, we find that the major image degradation happens in I frame, such as 1.4dB degradation of Akiyo's. P/B frames are less changed because the watermark is embedded only in I frame. Besides, video sequences of Class A tend to degrade much more than Class B and Class C.

The normalized correlation (NC), which represents the similarity between the original and the extracted watermark, are shown in Table III. Video sequences are encoded and embedded with watermark, and then decoded to extract watermark, i.e. at first no attack is applied. All NCs are approaching 1, which means the extracted watermark is almost identical to the original. (NC=[-1,1].) The NC of Class B and Class C videos with relatively high motion or texture may be smaller than 1. They contain more information to be encoded, so the quantization step size constrained from the same bitrates may be larger.

For robustness test, the compression attack is applied. That is, these videos are decoded and re-encoded at a lower bit rate, such as 0.8Mbps and 0.5Mbps. The results that NC after compression attacks are shown in Table IV. For Class A video sequences, the NC are still approaching 1. Only Foreman and Stefan re-compressed at 0.5Mbps get the moderately low NC. Then we also examine the effect of the frame weighting by the compression attack. First, we try to find out the proper constant a which controls the influence of the frame weighting. In Fig. 6, the x-axis represents the constant a, and the origin of the coordinates, X, means the voting procedure for block polarity does not applied. In other words, the watermark is extracted only from I frame. We can conclude that the constant a smaller than 1/7 does improve the accuracy of watermark extraction according to this figure. Therefore, we compare the NC with and without voting procedure, with the constant a set to 1/7. The results are shown in Fig. 7 and Fig. 8. With the voting procedure for the block polarity, the watermark actually gets more accurate NC than the one extracted only from the intra-frame, especial for the video sequence with high motion or texture.

Class A		Class B/C	
Akiyo	1	Foreman	0.9925
Mother & daughter	1	News	1
Hall_monitor	1	Stefan	0.9725

Table III The average NC of watermarks extracted from different video sequences without attack

Class A	0.5Mbps	0.8Mbps	Class B/C	0.5Mbps	0.8Mbps
Akiyo	0.97970	0.99719	Foreman	*0.69218*	0.88719
Mother & daughter	0.98939	0.99875	News	0.88478	0.97788
Hall_monitor	0.90346	0.94062	Stefan	*0.71156*	0.80720

Table IV The average NC of watermarks extracted from different video sequences with compression attacks

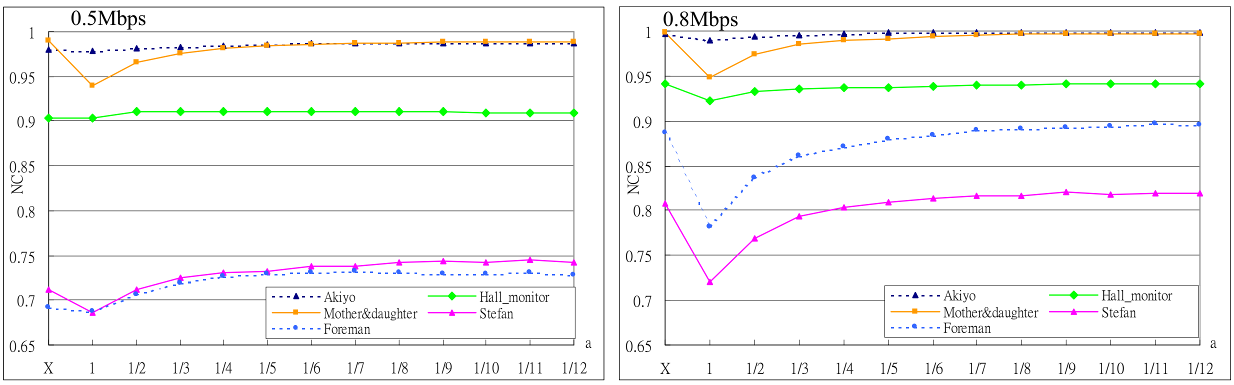

Fig. 6 The average NC vs. the constant a in the frame weighting (Compression attack at 0.5M and 0.8Mbps)

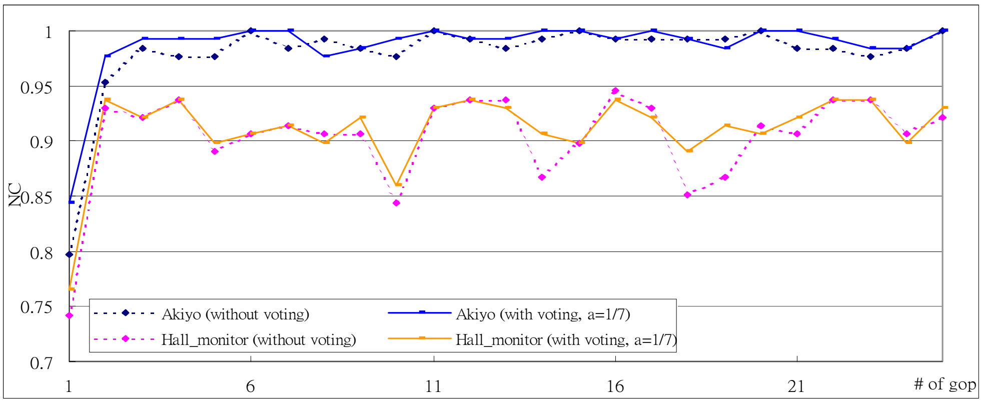

Fig. 7 NC comparison of with/without voting for Akiyo and Hall_monitor with compression attack at 0.5Mbps.

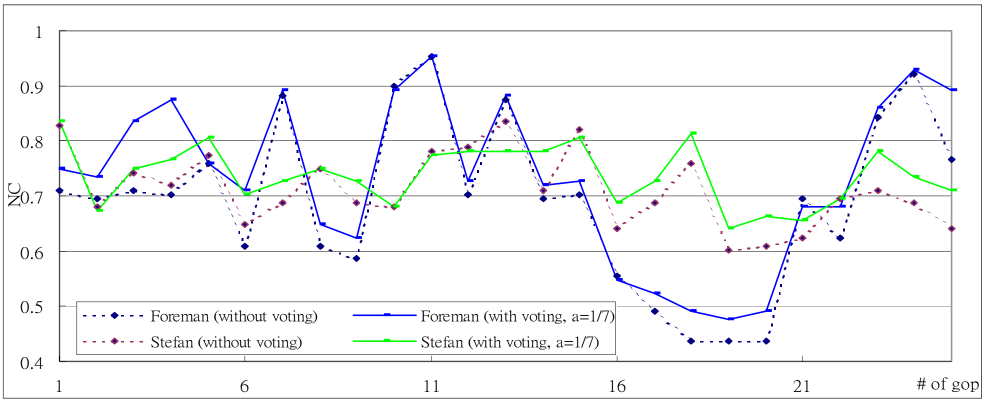

Fig. 8 NC comparison of with/without voting for Foreman and Stefan with compression attack at 0.5Mbps.

5. CONCLUSIONS

We have proposed a key-based video watermarking system on MPEG-2. This system can embed watermark and compress video at the same time. In addition, the watermark is embedded only in low frequency DCT coefficients of the intra-frame but can be extracted more accurately by referring to information in the inter-frames. The system possesses great imperceptibility that the PSNRs of the watermarked frames are almost the same as the un-watermarked ones. The simulation results show that the voting procedure for the block polarity does improve the NC of the extracted watermark.

6. REFERENCES

1. C. I. Podilchuk and E. J. Delp, "Digital watermarking: algorithms and applications," *IEEE Signal Processing Magazine*, vol. 18, no. 4, pp. 33-46, July 2001.
2. F. Hartung and M. Kutter, "Multimedia watermarking techniques," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1079-1107, July 1999.
3. C. T. Hsu and J. L. Wu, "DCT-based watermarking for video," *IEEE Transactions on Consumer Electronics*, vol. 44, no. 1, pp. 206-216, Feb. 1998.
4. T. Y. Chung, M. S. Hong, Y. N. Oh, D. H. Shin, and S. H. Park, "Digital watermarking for copyright protection of MPEG2 compressed video," *IEEE Transactions on Consumer Electronics*, vol. 44, no.3, pp. 895-901, Aug. 1998.

5. M. Barni, F. Bartolini, R. Caldelli, V. Cappellini, A. De Rosa, and A. Piva, "Robust frame-based watermarking for digital video," *12th International Workshop on Database and Expert Systems Applications*, pp. 825-829, 2001.
6. F. Hartung and B. Girod, "Digital watermarking of uncompressed and uncompressed video," *Signal Processing*, vol. 66, no. 3, pp. 283-301, 1998.
7. J. Zhang, H. Maitre, J. Li, and L. Zhang, "Embedding watermark in MPEG video sequence," *IEEE Fourth Workshop on Multimedia Signal Processing*, pp. 535-540, 2001.
8. J. Zhang, J. Li, and L. Zhang, "Video watermark technique in motion vector," *Proceedings of XIV Brazilian Symposium on Computer Graphics and Image Processing*, pp. 179-182, Oct. 2001.
9. G. C. Langelaar, R. L. Lagendijk, and J. Biemond, "Real-Time labeling of MPEG-2 compressed video," *Journal of Visual Communication and Image Representation*, vol. 9, no. 4, pp. 256-270, 1998.
10. I. J. Cox , M. L. Miller , and J.A. Bloom, *Digital watermarking*, Morgan Kaufmann, 2002.
11. ISO/IEC TTC1/SC29/WG11 13818-2 Information technology– Generic coding of moving pictures and associated audio information: video, 1995.
12. T.T. Lu, L.L. Lee, and P.C. Chang, "A key-based image watermarking system using subblock composition in DCT domain," *Proc. Multimedia Technology and Applications Conference*, pp. 157-163, Irvine, CA, Nov. 2001.