A Robust Video Watermarking Scheme of H.264

Shwu-Huey Yen[1] Mei-Chueh Chang[2] Chia-Jen Wang[3]
PRIA Lab., Department of Computer Science and Information Engineering
Tamkang University, Taipei, Taiwan, R.O.C.
{[1]105390,[2]changmc, [3]121718}@mail.tku.edu.tw

Abstract—**We present a video watermarking scheme which is robust to video coding standards H.264. The proposed scheme embeds watermark on all AC coefficients of 4×4 DCT blocks of I-frames. It integrates Watson's human visual system and the contents of 4×4 DCT blocks to determine the embedding strength. Since the high correlation of the video content within one scene, producing similar embedding strength, the scheme has the property of same watermarks for same scenes and different watermarks for different scenes. Thus, it can successfully resist the collusion attacks. Experiments validate that the proposed scheme has advantage of the invisibility, and it is robust to collusion attacks as well as H.264 compression.**

Keywords- video watermarking; human visual system (HVS); H.264

I. INTRODUCTION

As an effective method for copyright protection and content integrity verification of intellectual property, digital watermarking has become a very active area in the field of multimedia security. H.264 is the latest video compression standard and is employed in various video media such as HDTV, Blue-ray Disk, so it requires an appropriate watermarking technology.

In general, watermark can be embedded in compressed domain or uncompressed domain. The former has the benefit of time efficiency but with cost of failing to extract watermark once the video is recompressed by other standards. On the contrary, the latter has the benefit that watermarked video sequences can be compressed with different standards with the cost of decompression. There are many existing watermarking methods for compressed domain. Cao et al. [1] embedded watermark into residual blocks from intra-prediction of H.264 I-frame encoding. Noorkami et al. [2] embedded watermark in the quantized AC coefficients of I frames. Pröfrock et al. [3] used skipped macroblocks of H.264 compressed videos to embed the watermark. Noorkami et al. [4] [5] extended the human visual model (HVS) [6] [7] from 8×8 to 4×4 discrete cosine transform (DCT) block and embedded watermark into AC coefficient of residual blocks. Although they embedded watermark in compressed domain, but the embedding strength is decided in uncompressed domain. Lu et al. [8] used block polarity and index modulation to embed watermark into the quantized AC coefficients of I-frames. Generally speaking, embedding watermark in compressed domain needs a default quantized value. The quantized value is in conflict with robustness and it needs to discuss how to decide this value. Furthermore, if watermarked video frames are smoothed or sharpened, it may cause the intra-prediction modes on I-frames to be different from the original modes. The possibility of yielding different residual blocks makes methods using information of residual blocks to be hazardous.

Wu et al. [9] embedded watermark into I-frame of H.264 video. One pair of predicted DCT coefficients within the 4×4 blocks are used to embed 1-bit watermark information. Our simulation shows it is not robust in embedding location. Pröfrock et al. [10] claimed that watermark should be embedded in perceptual visible part of a video to avoid being suppressed during the compression process. They defined normed centre of gravity (NCG) to describe object borders and modified the NCG x, y-coordinates for watermark embedding. The main problem of the method is how to prevent the visible artifacts.

In this paper, we propose a watermarking scheme for H.264 on uncompressed domain such that it is transparent and robust. The paper is organized as follows. A brief background about watermarking system is described in Section II. The propose scheme is explained in Section III. Experimental results are presented in Section IV. Conclusions and future works are given in Section V.

II. RELATED WORKS

Human visual system (HVS) for 4×4 DCT blocks and the classification of 4×4 blocks are introduced in the following.

A. Human visual system

Watson [7] proposed an HVS for 8×8 DCT blocks. The human visual sensitivity for each DCT coefficient is decided by sensitivity function, luminance and contrast masking. The sensitivity function is defined as in Table I.

TABLE I. THE 8×8 DCT FREQUENCY SENSITIVITY TABLE

1.40	1.01	1.16	1.66	2.40	3.43	4.79	6.56
1.01	1.45	1.32	1.52	2.00	2.71	3.67	4.93
1.16	1.32	2.24	2.59	2.98	3.64	4.60	5.88
1.66	1.52	2.59	3.77	4.55	5.30	6.28	7.60
2.40	2.00	2.98	4.55	6.15	7.46	8.71	10.17
3.43	2.71	3.64	5.30	7.46	9.62	11.58	13.51
4.79	3.67	4.60	6.28	8.71	11.58	14.50	17.29
6.56	4.93	5.88	7.60	10.17	13.51	17.29	21.15

Since luminance and contrast affect the visual sensitivity, Watson redefined the threshold as

$$s[i,j,k] = \max\{t_L[i,j,k],\ |C[i,j,k]|^{w[i,j]}\, t_L[i,j,k]^{1-w[i,j]}\} \tag{1}$$

with

$$t_L[i,j,k] = t[i,j](C[0,0,k]/C_{0,0})^{\alpha_T} \tag{2}$$

where $[i,j,k]$ is the $[i,j]$ position of block k, $t[i,j]$ is the value on Table I, C[0,0,k] and $C_{0,0}$ are DC coefficients corresponding to block k and the mean luminance of the frame, α_T = 0.649 and $w[i, j]$=0.7 as suggested in [7].

In [5], authors observed the relation among basis functions of 4×4 DCT and 8×8 DCT. The basis functions of 4×4 DCT and 8×8 DCT are given below:

$$c_{i_4} c_{j_4} \cos(\frac{\pi(2n_1+1)i_4}{2\times 4})\cos(\frac{\pi(2n_2+1)j_4}{2\times 4}), \quad 0 \le i_4, j_4 \le 3, \tag{3}$$

$$c_{i_8} c_{j_8} \cos(\frac{\pi(2n_1+1)i_8}{2\times 8})\cos(\frac{\pi(2n_2+1)j_8}{2\times 8}), \quad 0 \le i_8, j_8 \le 7. \tag{4}$$

When $i_8 = 2\times i_4$ and $j_8 = 2\times j_4$ have the same frequency for i_4 & j_4 in 4×4 DCT and i_8 & j_8 in 8×8 DCT. Therefore, they modified Table I for 4×4 DCT matrices [6]. We combine the conclusion and Watson's luminance/contrast masking as the embedding strength of the watermark explained in the B of Section III.

B. Block classification

Chung et al.[11] performed 8×8 DCT on I-frames and each block is further classified with respect to its energy distribution into smooth, horizontal-edged, vertical-edged, diagonal-edged, and detailed blocks. Inspired by [11], we apply the similar approach to classify 4×4 blocks. For a given I-frame, first perform 4×4 DCT. Next, for each block, calculate B_{sum}, H_{sum}, V_{sum}, D_{sum}, and classify the block into flat, horizontal-edged, vertical-edged, diagonal-edged, detailed blocks following the procedure in Fig. 2. *coef*(i,j) is the DCT coefficient in (i,j) position and *mask*(i,j) is the value on the corresponding mask (Fig.1) in (i,j) position and the threshold Th_1 and Th_2 are set to be 50 and 90.

(a)

0	0	0	0
1	0	0	0
1	0	0	0
1	0	0	0

(b)

0	1	1	1
0	0	0	0
0	0	0	0
0	0	0	0

(c)

0	0.75	0	0
0.75	1	0.25	0
0	0.25	0	0
0	0	0	0

(d)

0	0.5	0.5	0
0.5	1	0	0
0.5	0	0	0
0	0	0	0

Figure 1. Masks for 4x4 block classification,
(a) *h_edge_mask* for horizontal edge, (b) *v_edge_mask* for vertical edge,
(c) *d_edge_mask* for diagonal edge, and (d) detailed-block mask.

If ($B_{sum} < Th_1$), class = flat-block
Else if ((E = Max (H_{sum} , V_{sum} , D_{sum})) > Th_2)
class = Dir (E)
Else class = detailed- block
where

$$B_{sum} = (\sum_{i=0}^{3}\sum_{j=0}^{3}| coef(i,j) |) - coef(0,0),$$

$$H_{sum} = \sum_{i=0}^{3}\sum_{j=0}^{3}| coef(i,j)\times H\_edge\_mask(i,j)|,$$

$$V_{sum} = \sum_{i=0}^{3}\sum_{j=0}^{3}| coef(i,j)\times V\_edge\_mask(i,j)|,$$

$$D_{sum} = \sum_{i=0}^{3}\sum_{j=0}^{3}| coef(i,j)\times D\_edge\_mask(i,j)|,$$

and *coef*(i,j) is the DCT coefficient in a block.

Figure 2. Procedure for 4x4 blocks classification

The purposes of masks are two-folds. One is the above-mentioned for classifying blocks, and the other is for embedding strength of the watermark. When a block has been classified, for example, as vertical-edged, it indicates the energies in (0,1), (0,2), (0,3) positions are large and the embedding strength should be small to preserve visual quality. And, for robustness consideration, the rest of positions can have larger embedding strength. As for a flat-block, since human visual is sensitive to smooth area, a mask is designed, shown in Fig. 3, which has all AC positions to be 1 indicating these positions are not suitable for embedding.

0	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

Figure 3. The mask for 4x4 flat blocks

III. PROPOSED METHOD

In this section, we describe the generation of the embedding watermark and the proposed scheme. The outlines are depicted in Fig. 4 and 5.

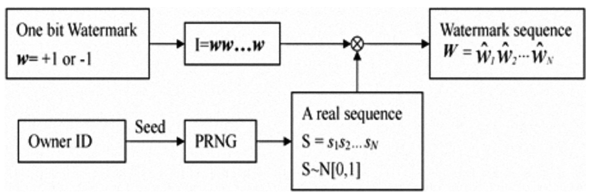
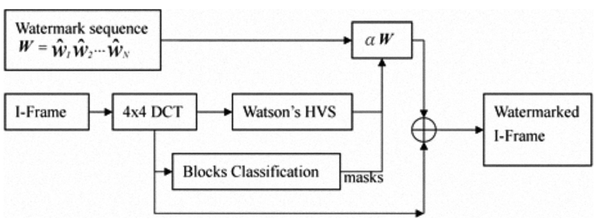


Figure 4. The generation of the watermark sequence



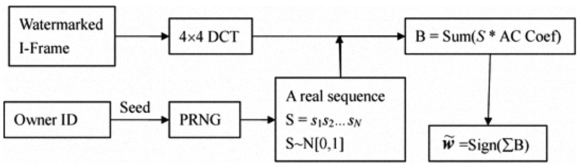


Figure 5. Block diagram for watermark embedding (*top*) and extracting (*bottom*)

A. *Generation of the watermark*

H.264 uses 16×16 or 4×4 intra-prediction on I-frames to remove spatial redundancy for better compression purpose. After intra-prediction, most of the residual energies on DCT blocks become small, and compress further result the majority to be 0. Thus, it is not reliable if watermark are only embedded into some coefficients of DCT. Besides, to prevent pooling effect from embedding watermark, it should embed watermark into different blocks and different frequencies. These make spread spectrum watermark to be a good choice.

In the proposed scheme, we embed one bit watermark $\boldsymbol{w}$, 1 or -1, in one I-frame. This one bit watermark $\boldsymbol{w}$ is then expanded into a sequence I of length N, all 1 or -1 same with $\boldsymbol{w}$. N =15r , r is the amount of 4×4 blocks in I-frame, 15 is the number of AC coefficients in one block. Through a pseudo random number generator (PRNG) and a meaningful seed number, such as owner ID, a real sequence S of length N is generated such that S ~ N[0,1]. The final watermark sequence $\boldsymbol{W}$ is obtained through an exclusive-like element-wise multiplication of S and I by

$$\boldsymbol{W} = S \otimes I = \hat{w}_1 \hat{w}_2 ... \hat{w}_N, \tag{5}$$

where ⊗ is an exclusive-like multiplication operation, i.e. if both operands have the same sign then the result is the multiplication of two operands but with negative sign, and positive sign otherwise. The sign relation is shown in Table II. Notice that sign of $S \otimes S$ is negative. Therefore,

$$\begin{aligned} sign(\boldsymbol{W} \otimes S) &= sign((S \otimes I) \otimes S) \\ &= sign((S \otimes S) \otimes I) = sign(I) = \boldsymbol{w}, \end{aligned} \tag{6}$$

where $sign(y)$ = +1, 0, -1 if y is positive, zero, or negative respectively. Moreover, $I = \boldsymbol{ww}...\boldsymbol{w}$ is a sequence of length N where $\boldsymbol{w}$ = +1 or -1, thus $sign(I)$ is defined as +1 (or -1) if $\boldsymbol{w}$ = +1 (or -1), i.e., $sign(I) = \boldsymbol{w}$. In extracting phase, we can perform ⊗ operation to S and the obtained coefficients where $\boldsymbol{W}$ is embedded. Thus the one bit watermark $\boldsymbol{w}$ can be extracted from the sign of I, as in Eq. (6).

TABLE II. SIGN TABLE FOR ⊗ OPERATION OF *a* AND *b*

	sign(a⊗b)	+	0	–
		sign(b)		
sign(a)	+	–	0	+
	0	0	0	0
	–	+	0	–

B. *Watermark embedding*

Given an I-frame of total r 4×4 blocks, we embed the watermark sequence $\boldsymbol{W} = \hat{w}_1 \hat{w}_2 ... \hat{w}_N$ into all AC coefficients on each 4x4 DCT block sequentially where N = 15r. The embedding strength on [i,j] (≠[0,0]) position of the k^{th} block is determined by $s[i,j,k]$ and $m'[i,j]$. $s[i,j,k]$ is the Watson's HVS described in Eq. (2) and $m'[i,j] = 0.5*(1-m[i,j])$ where $m[i,j]$ is the value on the mask given in Fig. 1 & 2 in accordance with the classification of the k^{th} block. If watermark $\hat{w}_n$ is to be embedded to the AC coefficient $x[i,j,k]$, $n \in \{1, 2, ..., N\}$, then

$$x'[i,j,k] = x[i,j,k] + (m'[i,j] \cdot s[i,j,k] + \beta) \cdot \hat{w}_n, \tag{7}$$

where β = 3, 2, 1, 0, if block k is classified into a detailed, diagonal- edged, vertical- or horizontal- edged, flat- block, respectively. Eq. (7) can be simplified as

$$x'[i,j,k] = x[i,j,k] + \alpha[i,j,k] \cdot \hat{w}_n, \tag{8}$$

where $\alpha[i,j,k]$ (≥ 0) is the embedding strength on position [i, j, k]. Observe that when block k is a flat-block, $\alpha[i,j,k]$ is zero, i.e., $x'[i,j,k] = x[i,j,k]$, and when block k is a detailed-block the embedding strength is generally large.

C. *Watermark extracting*

When a watermarked I-frame is received, we perform 4×4 DCT and order all AC coefficients sequentially as $\tilde{X} = \tilde{x}_1 \tilde{x}_2 ... \tilde{x}_N$ where N = 15r, r is the total number of 4×4 blocks in this frame. Generate the real sequence $S = s_1 s_2 ... s_N$ with the same seed number as in the embedding phase. Without loss of generality, let $\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_{15}$ be the AC coefficients on the k^{th} block, and $s_1 s_2 ... s_{15}$ be the corresponding subsequence of S. Define

$$B_k = \sum_{l=1}^{15} \tilde{x}_l \otimes s_l, \tag{9}$$

and

$$Total\_sum = \sum_{k=1}^{r} B_k. \tag{10}$$

Then the extracted watermark $\tilde{w}$ of this I-frame is the sign of $Total\_sum$, i.e.,

$$\tilde{w} = sign(Total\_sum). \tag{11}$$

Let $\tilde{x}_l = \tilde{x}[i,j,k]$ be the AC coefficient on position [i,j,k], then with Eq.(8) the multiplication in Eq. (9) becomes

$$\begin{aligned} \tilde{x}[i,j,k] \otimes s_l &= (x[i,j,k] + \alpha[i,j,k]\hat{w}_l) \otimes s_l \\ &= x[i,j,k] \otimes s_l + (\alpha[i,j,k]\hat{w}_l) \otimes s_l. \end{aligned} \tag{12}$$

According to [12], the distribution of AC coefficients of DCT is approximately Cauchy distribution which is symmetric with zero mean, and the sequence S is also symmetric with zero mean. Therefore, a lot of the first term on the right side of Eq. (12) are cancelled out in the summation of Eq. (10). The second term on the right side of Eq. (12), $\alpha[i, j, k] \geq 0$ and from Eq. (6), their summation in Eq. (10) is a large number with the sign similar to $\boldsymbol{w}$. The watermark is consequently extracted via the sign of *Total_sum*.

IV. EXPERIMENTAL RESULTS

Various experiments are conducted to verify the proposed method. In the following, video sequences are compressed under different QP values to illustrate the robustness of the method, and collusion attacks are simulated too.

A. *Compression attacks*

In our experiments, the first 100 frames of five CIF video clips, *Akiyo*, *Foreman*, *Bus*, *Mobile* and *Tempete*, of 352×288 are employed. The version of H.264 used is JM 12.3 (FRExt). The sequence type is IBPBP where I-frames are on frames 0, 6, …, 96.

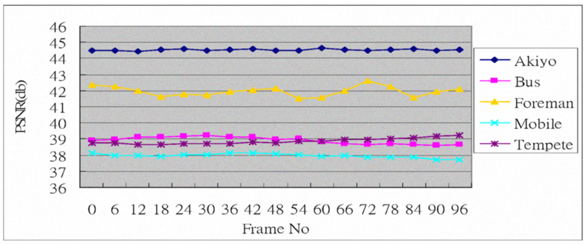


Figure 6. PSRN of watermarked videos

To testify the transparency of the proposed algorithm, Fig. 6 shows PSNR values after watermarked. As Observed, *Mobile* has the lowest PSNR value of about 38 due to highly textured content. Hence, most of its blocks are classified as detailed causing strong embedding strength. Likewise to *Bus* and *Tempete* with PSNR values approximately 39. On the other hand, *Akiyo* has relatively smooth texture, so many blocks are classified as flat blocks, and, thus, PSNR value is above 44. Overall, the proposed method satisfies transparency requirement.

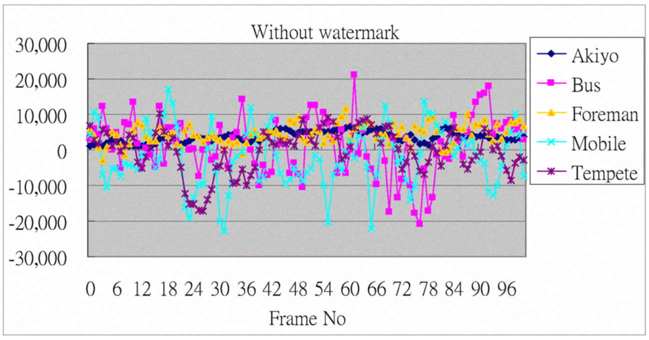


Figure 7. *Total_sum* obtained from video without watermarking

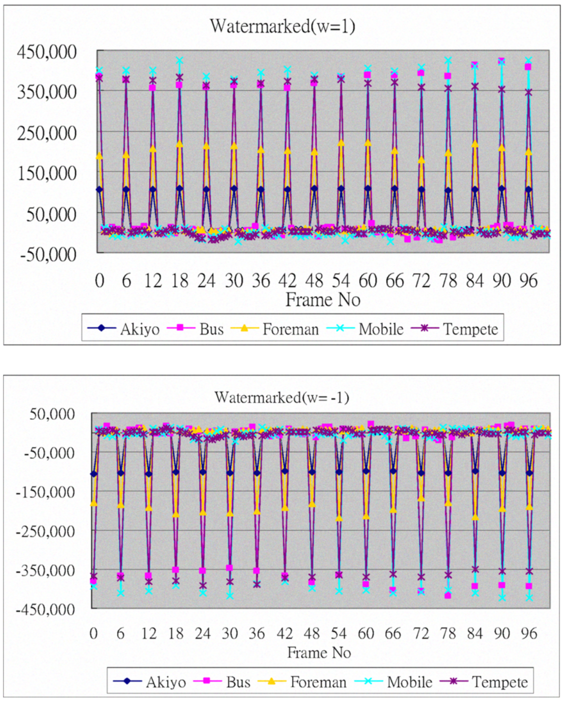


Figure 8. Extraction of the one bit watermark for $w = 1$ (*top*) and $w = -1$ (*bottom*)

To extract watermark, we calculate *Total_sum* as in Eq. (10). Fig. 7 shows the extraction result from unwatermarked video clips. The extracted energies *Total_sum* are quite random and show no significant magnitude on I-frames. After watermark embedding, as in Fig. 8, the extracted energies have distinct peaks on I-frames on positive or negative value conformable to the embedding $\boldsymbol{w}$. Comparing Fig. 7 & 8, we can easily determine whether a video is watermarked. In addition, *Total_sum* on watermarked I-frames of *Mobile* is around 400,000 due to highly textured content, and *Akiyo* is around 100,000 due to smooth content when $\boldsymbol{w} = 1$ is embedded. For the rest of experiments, $\boldsymbol{w} = 1$ is embedded whenever there is an embedding process.

To illustrate our method can resist the compression of H.264, all five video sequences are compressed under different QP values. Fig. 9 shows the result of *Akiyo*. It shows extraction energies of unwatermarked video (blue: not compressed, pink: compressed QP=28). Both curves show small values without peaks which implies that compression or not has no effect on unwatermarked frames. It also shows extraction energies of watermarked I-frames with compression QP = 28, 32, 36, 40. When QP = 28, the extraction of energies on I-frames exhibit distinct peaks. The higher value of QP is, the lower peaks of the extraction energies are. When QP is 40, the extraction energy is approximately 15% of that in QP = 28. Nevertheless, the extraction energy still shows peaks on those watermarked frames. Fig. 10 shows extraction result of compression. Due to limited space, only some video sequences with compression QP=40 are shown. The extraction energies exhibit peaks clearly on I-frames. The experiments confirm that our method can resist compression attacks.

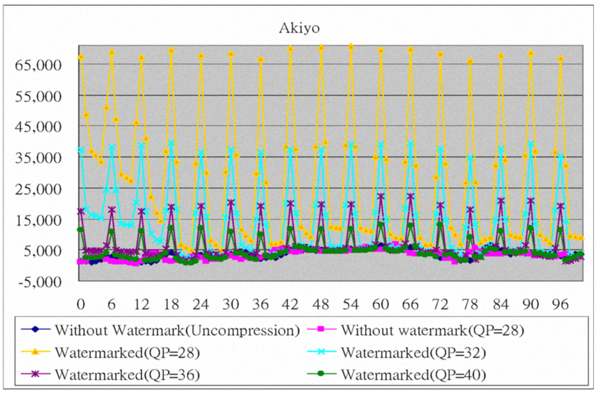


Figure 9. Comparison for before/after of compression (different QP) and embedding

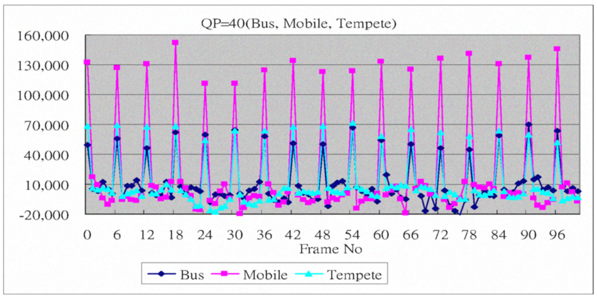


Figure 10. Extraction energies of compressed watermarked videos under QP 40

B. *Collusion attacks*

As video sequence contains enormous similar frames, that open new opportunities for collusion. In [13][14], authors addressed two types of collusion, TFA (Temporal Frame Average) and WER (Watermark Estimation Remodulation).

- **TFA attack**

To simulate TFA attacks, we perform temporal low-pass filtering of size 3×3, i.e. $f'(k) = 1/3(f(k\text{-}1)+ f(k)+ f(k+1))$. The attacked frames $f'(k)$ are shown in Table III. Since *Akiyo* has less motion, the attacked frames are almost visual imperceptible. *Foreman* has head motion causing the video quality degraded. From Fig. 11, affected by temporal averaging, the extraction energies are reduced. In addition to the peaks on I-frames, we observe that the adjacent frames to I-frame also have high energies. But for the purpose of verifying the existence of the watermark, they still exhibit a regular pattern enough to claim the existence of the watermark.

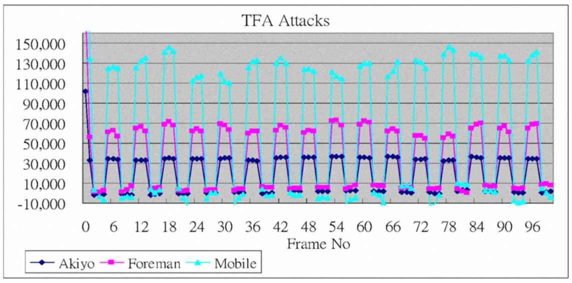


Figure 11. Energy extraction on TFA attacked videos

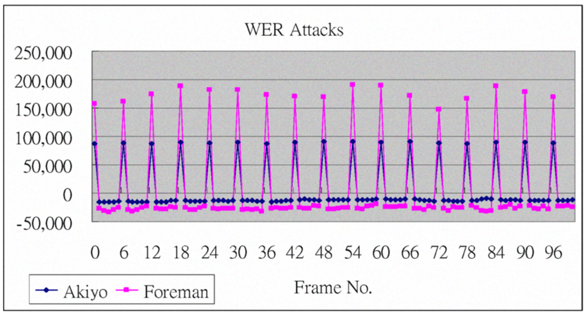


Figure 12. Energy extraction on WER attack (without knowing where watermark is embedded)

TABLE III. RESULTS OF TFA ATTACKS

Akiyo	 K=14	 K=35	 K=76
Mobile	 K=4	 K=12	 K=63
Foreman	 K=1	 K=70	 K=90

- **WER attack**

In WER attack, a rough estimated watermark ***W'*** is obtained from every watermarked frame. Then average all rough estimated watermarks to get the final estimated watermark ***W''***. Finally, subtract the final estimated watermark from watermarked frames to erase the watermark. In the simulation, regarding watermark as a noise, a rough estimated watermark is obtained by subtracting a low pass filtered watermarked frame from its original. Fig. 12 shows the extraction results of attacked videos. These curves show that the WER attack is not successful in erasing the watermark.

V. CONCLUSIONS AND FUTURE WORKS

In this paper, we propose a spread spectrum watermarking scheme such that embedding strength is based on human visual system as well as video content. The extraction needs only the seed number without using other information such like the original video. The experiments show the method can resist various QP of H.264 compression as well as the collusion attacks.

As in the experiments demonstrated, the extraction energy clearly demonstrates whether a video frame is watermarked. In the future, we hope to develop an automatic way to determine the existence of the watermark instead of inspection. Besides, we will extend the scheme on compressed domain to realize a real time watermarking system.

REFERENCES

[1] Cao, H., Zhou, J., Yu, S., "An implement of fast hiding data into H.264 bitstream based on intra-prediction coding," *Proceedings of SPIE*, Vol. 6043, 2005, pp. 604301.1-604301.8

[2] Noorkami, M., Mersereau, R. M., "Compressed-domain video watermarking for H.264," *IEEE Int'l Conf. on Inage Processing* (ICIP 2005), 2005, pp. 890-893.

[3] Pröfrock, D., Richter, H., Schlauweg, M., Müller, E., "H.264/AVC video authentication using skipped macroblocks for an erasable watermark," *Proceedings of SPIE*, Vol. 5960, 2005, pp. 1480-1489.

[4] Noorkami, M., Mersereau, R. M., "Towards robust compressed-domain video watermarking for H.264," *Proceedings of SPIE*, Vol. 6072, 2006.

[5] Noorkami, M., Mersereau, R. M., "A framework for robust watermarking of H.264-encoded video with controllable detection performance," *IEEE Transactions on Information Forensics and Security*, Vol. 2, 2007, pp.14-23.

[6] Peterson, H. A., Peng, H., Morgan, J., Pennebaker, W., "Quantization of color image components in the DCT domain," *Proceedings of SPIE*, Vol. 1453, 1991, pp. 210–222.

[7] Watson, A., "DCT quantization matrices visually optimized for individual images," *Proceedings of SPIE*, Vol. 1913, 1993, pp.202-216.

[8] Lu, T.-T., Hsu, W.-L., Chang, P.-C., "Blind video watermarking for H.264," *Conf. in Electrical and Computer Engineering* (CCECE'06), 2006, pp. 2353-2356.

[9] Wu, G.-Z., Wang, Y.-J., Hsu, W.-H., "Robust watermark embedding detection algorithm for H.264 video," *Journal of Electronic Imaging*, Vol. 14, 2005, 013013.

[10] Pröfrock, D., Schlauweg, M., Müller, E., "A new uncompressed-domain video watermarking approach robust to H.264/AVC compression," *Proceedings of the Third IASTED International Conference on Signal Processing, Pattern Recognition, and Applications*, 2006, pp. 99-104.

[11] Chung, T.-Y., Hong, M.-S., Oh, Y.-N., Shin, D.-H., Park, S.-H., "Digital watermarking for copyright protection of MPEG2 compressed video," *IEEE Transactions on Consumer Electronics*, Vol. 44, 1998, pp. 895-901.

[12] Altunbasak, Y., Kamaci, N., "An Analysis of the DCT Coefficient Distribution with the H.264 Video Coder," in Proc. ICASSP'04, Montreal, Canada,pp.177-180 (2004)

[13] Doërr, G., Dugelay, J.-L., "A guide tour of video watermarking," *Signal Processing: Image Communication* 18, 2003, pp. 263–282.

[14] Doërr, G., Dugelay, J.-L., "Security pitfalls of frame-by-frame approaches to video watermarking," *IEEE Transactions on Signal Processing*, Vol. 52, 2004, pp.2955-2964.