10

Video Watermarking Technique using Visual Sensibility and Motion Vector

Mariko Nakano-Miyatake and Hector Perez-Meana
National Polytechnic Institute of Mexico
Mexico

1. Introduction

Together with the rapid growth of Internet service, copyright violation problems, such as unauthorized duplication and alteration of digital materials, have increased considerably (Langelaar et al., 2001). Therefore copyright protection over the digital materials is a very important issue that requires an urgent solution. The watermarking is considered as a viable technique to solve this problem. Until now, numerous watermarking algorithms have been proposed. Most of them are image watermarking algorithms and relatively few of them are related with video sequences. Although image watermarking algorithms can be used to protect the video signal, generally they are not efficient for this purpose, because image watermarking algorithms does not consider neither temporal redundancy of the video signal nor temporal attacks, which are efficient attacks against video watermarking (Swanson et al., 1998).

Generally, in the watermarking schemes for copyright protection, the embedded watermark signal must be imperceptible and robust against common attacks, such as lossy compression, cropping, noise contamination and filtering (Wolfgang et al., 1999). In addition, video watermarking algorithms must satisfy the following requirements: a blind detection, high speed process and conservation of video file size. The blind detection means that the watermark detection process does not require original video sequence, and the temporal complexity of watermark detection must not affect video decoding time. Also the file size of video sequence must be similar, before and after watermarking. Due to the redundancy of the video sequence, some attacks such as frame dropping and frame averaging can effectively destroy the embedded watermark, without cause any degradation to the video signal. A design of an efficient video watermarking algorithm must consider this type of attacks (Wolfgang et al., 1999).

Basically, video watermarking algorithms can be classified into three categories: watermarking in base band (Wolfgang et al., 1999; Hartung & Girod 1998; Swanson et al., 1998; Kong et al., 2006), watermarking during video coding process (Liu et al., 2004; Zhao et al., 2003; Ueno 2004; Noorkami & Mersereau 2006) and watermarking in coded video sequence (Wang et al., 2004; Biswas et al., 2005; Langelaar & Lagendijk 2002). In the base band technique, the watermarking process is realized in uncompressed video stream, in which almost all image watermarking algorithms can be used, however generally computational complexity for watermark embedding and detection is considerably high for

Source: Visual Servoing, Book edited by: Rong-Fong Fung,
ISBN 978-953-307-095-7, pp. 234, April 2010, INTECH, Croatia, downloaded from SCIYO.COM

its practical use. In the algorithm proposed by Wolfgang et al. (1999), Just Noticeable Difference (JND) is used, in the Discrete Cosine Transform (DCT) domain, to determine an adequate watermark embedding energy. Hartung and Girod (1998) proposed an algorithm, in which binary data modulated by pseudo-random sequence is embedded into luminance component of each video frame. Swanson et al. (1998) proposed an algorithm based on the Discrete Wavelet Transform (DWT) through temporal sequences. Kang et al. applied singular value decomposition (SVD) to each frame of video data, and then embedded the watermark signal into the singular values.

The watermarking technique in compressed video data, embed the watermark signal into bit sequence compressed by standard coding, such as MPEG-2 and MPEG-4, etc. Generally this technique has lower computational cost, compared with other methods; however the number of watermark bits must be limited by compression rate. In the algorithm proposed by Wang et al. (2004), the watermark signal is embedded only into the I-frames using JND concept, while Biswas et al. (2005) directly embedded the watermark signal into MPEG compressed video sequence, modifying DCT coefficients. Also in the algorithm proposed by Langelaar and Lagendik (2001), the watermark signal is embedded into the I-frames in the DCT domain.

The watermarking algorithms operating during MPEG coding process are inherently robust against standard compression attacks, without increase the compression rate of the video sequence. Liu et al. (2004) proposed an algorithm, where the watermark signal is embedded into the motion vectors, and using the watermarked motion vectors, MPEG bit sequence is generated. While Zhao et al. (2003) proposed a fast algorithm to estimate motion vectors during the compression process, and also they embed the watermark signal, modifying angle and magnitude of the motion vectors. In the algorithm proposed by Ueno (2004), motion vectors are used to determine an adequate position in DCT coefficients of I-frames for watermark embedding. Noorkami and Mersereau (2006) estimated motion regions, computing spatial distribution of motion through several consecutive frames. Large amount of watermark bits are embedded into dynamic motion regions, while small amount of watermark bits are embedded into statistic regions. In this manner the artifact caused by watermark embedding can be avoided (Noorkami & Mersereau 2006).

In this paper, a video watermarking algorithm is proposed, in which watermark embedding is carried out during MPEG2 coding process. The proposed algorithm uses three criteria based on deficiency of the Human Visual System (HVS) to embed robust watermark, while preserving its imperceptibility. First criterion is based on difference of sensibility of the HVS to basic three color channels (red, green and blue), and second one is based on frequency masking of the HVS proposed by Tong and Venetsanopoulos (1998). Third criterion is based on deficiency of the HVS to trace high speed motion region, which is related directly to the motion vector of each macro-block. The third criterion is only applied to P-frames, while other two criteria are applied both I-frames and P-frames. In the proposed algorithm, B-frames are excluded from the watermark embedding and detection process to reduce computational complexity. In this manner, watermark embedding and detection processes don't cause any delay in coding and decoding processes. Simulation results show the watermark imperceptibility and robustness against common signal processing and some intentional video frame attacks, such as frame dropping, frame averaging and frame swapping. The watermark imperceptibility is measured using the Peak Signal Noise Ratio (PSNR) and a HVS based objective evaluation proposed by Wang and Bovik (2004).

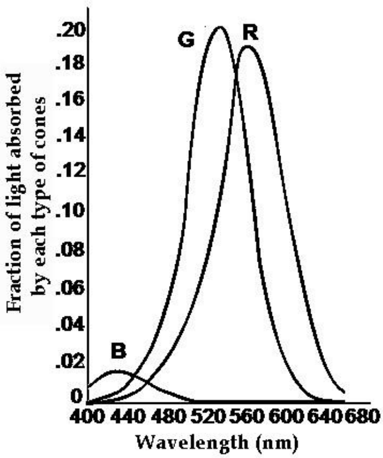

Fig. 1. Sensibility of HVS to different wavelength related to three basic colors.

2. Proposed system

In this section, a detailed description of the proposed video watermarking algorithms is provided.

2.1 HVS based criteria

In the proposed algorithm, the three criteria mentioned in introduction are used to embed imperceptible and robust watermark into a video sequence. These criteria are based on deficiency of sensibility of the HVS to blue channel, regions with details, such as texture region and region with high motion speed.
In the HVS, there are three types of cones that react to the basic three colors: red, green and blue. The number of cones reacted to blue is 30 times smaller than the number of cones reacted to red or green, which means the HVS has deficiency of sensibility to blue color (Sayood, 2000). The figure 1 shows fraction of light absorbed by each type of cone, here R, G, and B represent red, green and blue colors, respectively. The proposed algorithm embeds watermark signal into the blue channel, using its HVS deficiency. Generally color space used for video sequence is YUV or YCrCb, therefore firstly these color spaces are transformed into RBG color space using the transform matrix given by (1) and (2). (Plataniotis & Venesanopoulos 2000).

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix}^{-1} \begin{bmatrix} Y \\ C_r \\ C_b \end{bmatrix} \tag{1}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.5149 & -0.100 \end{bmatrix}^{-1} \begin{bmatrix} Y \\ U \\ V \end{bmatrix} \tag{2}$$

The second criterion is based on the difference of the HVS's sensibility to spatial features, such as texture, edge and plain regions. Each I-frame and P-frame is divided into blocks of size 8x8, and then 2D-DCT is applied to each block. Classification of each block is carried out using algorithm proposed by Tong and Venetsnopoulos (1998), which is described briefly as follows:

1. Each DCT block is divided into 4 areas denoted by DC, L, E and H, as shown by figure 2.
2. The sum of absolute value of coefficients belonging to DC, L, E and H are denoted as S_{DC}, S_L, S_E and S_H, respectively.
3. Using the following conditions, each block of DCT is classified as "edge block", "Texture block" or "plain block".

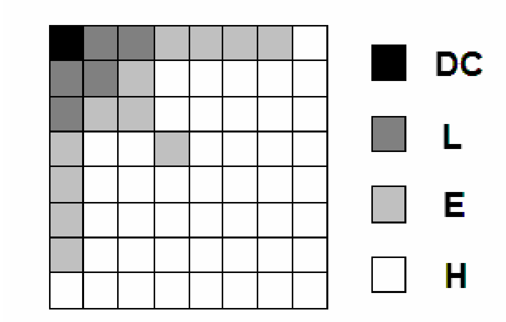

Fig. 2. Four regions of a DCT block

<u>Conditions for "edge block"</u>

If either of two conditions A or B is satisfied, then the DCT block is classified as "edge block".

Condition-A

$$\text{If } (S_E + S_H \le \mu_1) \wedge \left(\frac{S_L}{S_E} \ge \alpha_1\right) \wedge \left(\frac{S_L + S_E}{S_H} \ge \beta_1\right)$$
$$\vee$$
$$\text{If } (S_E + S_H \le \mu_1) \wedge \left(\frac{S_L}{S_E} \ge \beta_1\right) \wedge \left(\frac{S_L + S_E}{S_H} \ge \alpha_1\right)$$
$$\vee$$
$$\text{If } (S_E + S_H \le \mu_1) \wedge \left(\frac{S_L + S_E}{S_H} \ge \gamma\right).$$

Condition-B

$$\text{If } (S_E + S_H > \mu_1) \wedge \left(\frac{S_L}{S_E} \ge \alpha_2\right) \wedge \left(\frac{S_L + S_E}{S_H} \ge \beta_2\right)$$
$$\vee$$
$$\text{If } (S_E + S_H > \mu_1) \wedge \left(\frac{S_L}{S_E} \ge \beta_2\right) \wedge \left(\frac{S_L + S_E}{S_H} \ge \alpha_2\right)$$
$$\vee$$
$$\text{If } (S_E + S_H > \mu_1) \wedge \left(\frac{S_L + S_E}{S_H} \ge \gamma\right)$$

where the operations $\wedge$ and $\vee$ mean logical multiplication and logical addition, and six parameters used in the conditions are $\mu_1 = 900,\ \alpha_1 = 2.3, \beta_1 = 1.6, \alpha_2 = 1.4, \beta_2 = 1.1$ y $\gamma = 4$.

<u>Condition for "texture block"</u>

If the condition-A is not satisfied and $S_E + S_H > \kappa$, or if the condition-B is not satisfied then the block is classified as "texture block", where $\kappa = 290$.

Condition for "plain block"

If $S_E + S_H \le \mu_2$ is satisfied or the condition-A is not satisfied and $S_E + S_H \le \kappa$, then the block is classified as "plain block", where $\mu_2 = 125$.

Fig. 3. An example of the block classification using second criterion

The figure 3 shows an example of block classification using the above algorithm (Tong and Venetsanopoulos, 1998). Here black blocks, gray blocks and white blocks indicate "plain blocks", "texture blocks" and "edge blocks", respectively.

The last criterion is based on deficiency of the HVS to trace regions with high speed motion. Actually the MPEG coding uses this deficiency to reduce temporal redundancy of video sequence. The macro-blocks, whose motion vector has large magnitude, can be classified as regions with high speed motion. The macro-blocks classified as high motion speed regions are adequate to embed a high energy watermark signal without causing any visual distortion. The magnitude of the motion vector is computed by (3).

$$Mmv_i = \sqrt{mvh_i^2 + mvv_i^2}, \quad i = 1..MB \tag{3}$$

where mvh_i, mvv_i are horizontal and vertical components of the motion vector of i-th macro-block, and *MB* is total number of macro-blocks. To determine macro-blocks with high speed motion, a threshold value *Th_mv* is introduced, which value is computed by (4)

$$Th\_mv = \frac{1}{MB}\sum_{i=1}^{MB} Mmv_i \tag{4}$$

Using this value, macro-block is classified as follows.

If $Mmv_i < Th\_mv$ then i-th macro-block doesn't have motion (static region).

If $Mmv_i \ge Th\_mv$ then i-th macro-block has motion (dynamic region).

The macro-blocks, whose magnitude of motion vector is smaller than the threshold, are considered as static blocks and the motion vectors of the static blocks are ignored. The figure

4(a) and (b) show two consecutive frames, all motion vectors before classification are depicted in fig 4(c) and fig. 4(d) shows only the motion vectors classified as high speed motion.

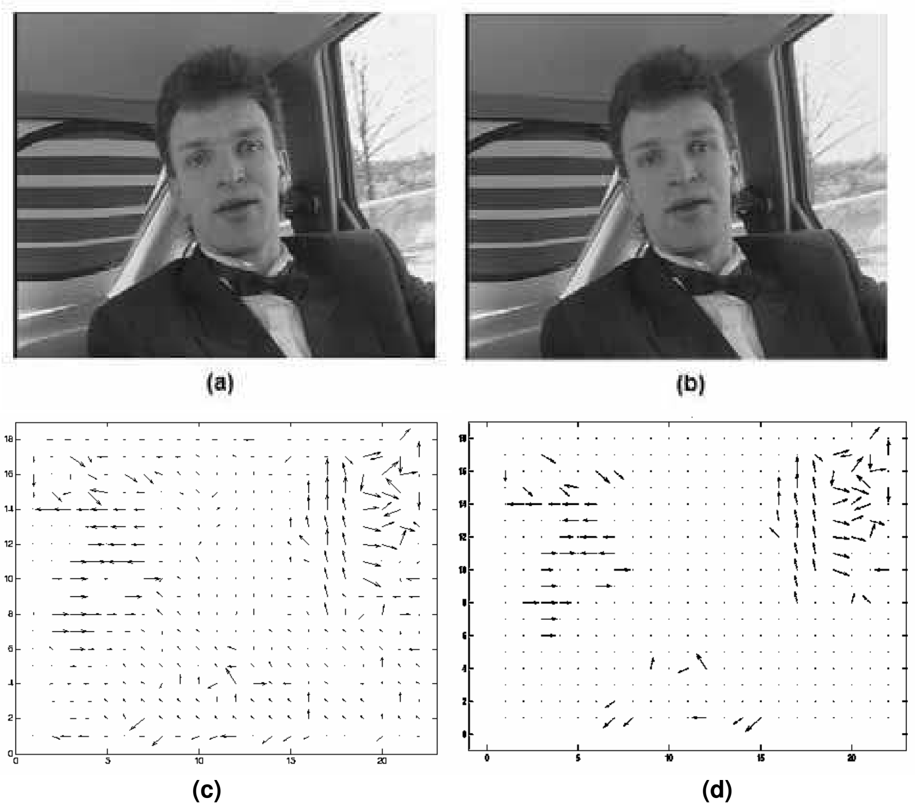

Fig. 4. (a) and (b) are two consecutive frames, (c) all motion vectors got from two consecutive frames and (d) motion vectors with high speed motion.

Energy	$B \in$ Plain	$B \in$ Texture	$B \in$ *Edge*
$B \in C_I$	0.4	0.6	0.9
$B \in C_P$ $B \notin Mb_{motion}$	0.4	0.6	0.9
$B \in C_P$ $B \in Mb_{motion}$	0.8	1.2	1.8

Table 1. Watermark embedding energy

Combining latter two criteria, the spatial feature of 8x8 blocks and the motion feature of macro-blocks (16x16), the watermark embedding energy for I-frames and P-frames is determined experimentally using 10 video sequences. The embedding energies of different type of blocks are shown by the table 1; in which, *B* means blocks of size 8x8 of I-frames and P-frames, and Mb_{motion} means macro-blocks with high speed motion. Each macro-block contains 4 blocks *B*, and C_I, C_P mean I-frames and P-frames, respectively. Figure 5 shows an example of block classification together with the watermark embedding energy assigned to each block.

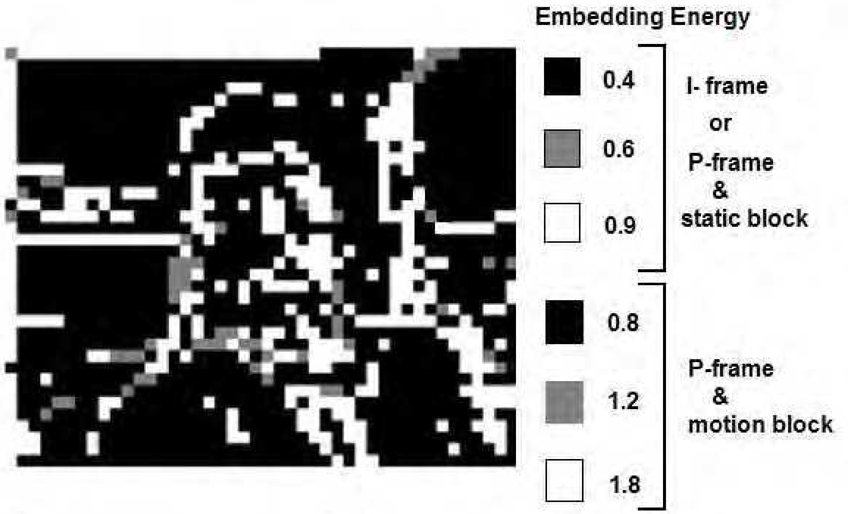

Fig. 5. An example of watermark embedding energy assignation

2.2 Watermark embedding process

The watermark embedding process of the proposed algorithm consists of two parts: first part and second part. In the first part, an adequate watermark embedding energy for each block is calculated, and in the second part, the watermark signal is embedded into each block using the adequate embedding energy computed in the first part. The video sequence is decomposed by RGB color space, and only the blue channel is used for watermark embedding. The blue channel is divided into blocks of 8x8 and then each block is transformed by 2D-DCT. Each block is classified into three categories: plain block, texture block and edge block. For P-frames, the macro-blocks are generated by combining four neighbor blocks of 8x8, and then each macro-block is classified between static block and block with high speed motion. Using table 1, the watermark embedding energy is assigned to each block (8x8). The watermark signal is a pseudo-random sequence generated by the secret user's key. Watermark embedding is performed by (5).

$$\begin{aligned} &DCT_k(i,j) = DCT_k(i,j) + \alpha_k \left|DCT_k(i,j)\right| W_k \\ &(i,j) = (1,2) \ \text{ for } C_I \\ &(i,j) \in \{(1,2),(1,3),(2,1),(2,2),(3,1)\} \ \text{ for } C_P \end{aligned} \tag{5}$$

where $DCT_k(i,j)$ is *(i,j)-th* DCT coefficient of *k-th* block and α_k is the embedding energy assigned to *k-th* block. For I-frames, a watermark bit is embedded only into a AC coefficient with lowest frequency, while for P-frames; five watermark bits are embedded into five lowest AC coefficients. The figure 6 shows the watermark embedding process.

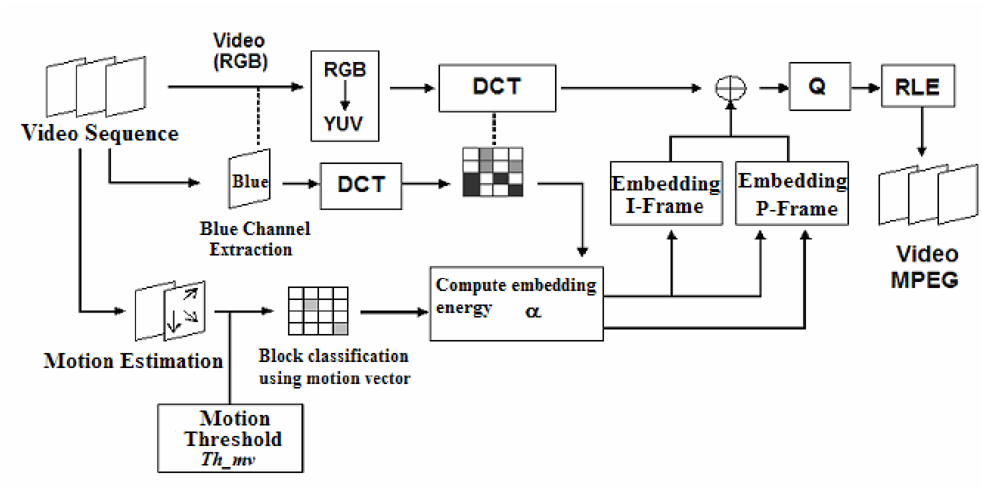

Fig. 6. Watermark embedding process

2.3 Watermark detection process

In the watermark detection process, only the blue channel of the watermarked and possibly distorted video sequences is used, which is divided into blocks of 8x8 pixels. 2D-DCT is applied to each block of I-frames and P-frames. The lowest AC coefficient of each block of I-frame and the five lowest coefficients of each block of P-frame are extracted. The extracted coefficients are concatenated through all blocks of I and P frames to generate an extracted watermark sequence Y. Finally to determine if owner's watermark is presented in the video sequence or not, the cross-correlation between the extracted watermarked coefficients Y and the owner's watermark sequence W, is calculated as shown by (6).

$$C = \frac{1}{L}\sum_{i=1}^{L} W_i Y_i \tag{6}$$

where L is watermark length.

If the cross-correlation value C is bigger than a predetermined threshold value Th_w, it is considered that the owner's watermark signal is presented in the video sequence; otherwise the video sequence was not watermarked or watermarked by another watermark sequence. Here the threshold value plays a very important role and this value is determined considering two probabilities: probability of detection error and probability of false alarm error. In the proposed algorithm, adaptive threshold value is used, which is given by (7). This threshold value guarantees that false alarm error probability is smaller than 10^{-6}. (Piva et al., 1997).

$$Th_w = \frac{1}{3L}\sum_{i=1}^{L} |Y_i| \alpha_i \tag{7}$$

3. Experimental results

To evaluate the proposed algorithm, several video sequences with format YUV-CIF are used. The figure 7 shows some video sequences used in the evaluation. The proposed algorithm is evaluated from embedded watermark imperceptibility and robustness points of view.

Fig. 7. Some video sequences used for evaluation of the proposed algorithm

3.1 Imperceptibility

Embedded watermark imperceptibility of the proposed algorithm is evaluated using PSNR and the universal quality index (UQI) proposed by Wang et al. (2004). The UQI is an objective quality assessment related with perceptual distortion, which was originally developed to assess a visual quality for images as given by (8). In order to assess perceptual quality of the video sequence, UQI value of each macro-block is compensated according to the motion speed of the macro-block.

$$UQI = \frac{4\sigma_{xy}\bar{x}\cdot\bar{y}}{\left(\sigma_x^2+\sigma_y^2\right)\left[\left(\bar{x}\right)^2+\left(\bar{y}\right)^2\right]} \quad (8)$$

where $\bar{x}, \bar{y}$ are mean values of original image x and processed image y, respectively, σ_x^2 and σ_y^2 are variances of x and y. respectively, and σ_{xy} is their covariance. The value range of UQI is [0, 1.0], when the video sequence under analysis is identical with the original one, the UQI value is equal to 1.0; and as the visual distortion increase, the UQI value decrease.
The figure 8 shows the original I-frame and P-frame, and their watermarked versions, respectively, together with the PSNR values. From this figure, we can considered that the embedded watermark is imperceptible, because the PSNR value is bigger than 40dB for I-frame, and it is approximately 40dB for P-frame, and UQI of the watermarked video sequence is equal to 0.96, which indicates that the embedded watermark is imperceptible by the HVS.

	Original frame	Watermarked frame	PSNR (dB)
I-frame			42.91
P-frame			39.83

Fig. 8. Watermark imperceptibility of I-frame and P-frame

3.2 Watermark robustness

To evaluate the watermark robustness of the proposed algorithm, the watermarked video sequences are attacked using common signal and image processing tasks, such as coding rate change, impulsive and Gaussian noise contamination, frame dropping, frame swapping, frame averaging and cropping. The figure 9 shows the watermark robustness against the coding rate change, applying quantized matrix with different quality factor during MPEG coding process. Fig 9(a) shows one frame of the watermarked video without any attack, fig. 9 (c) shows one frame of the watermarked and compressed video with quality factor equal to 30. Figure 9(b) and fig. 9(d) show the detector responses. From these figures, it is concluded that the embedded watermark is robust to high rate compression; actually the watermark signal survived after compression with quality factor of 30. If the watermarked video sequence is compressed using a lower quality factor than 30, the embedded watermark signal can be lost, however in this situation the distortion caused by compression is not acceptable and the attacked video sequence no longer has commercial value. In all robustness evaluations, the watermarked videos are analyzed using 1000 possible watermark signals generated by 1000 different keys; and the embedded watermark generated by the owner corresponds to the key equal to 450. In all figures, the horizontal line represents the threshold value calculated by (7).

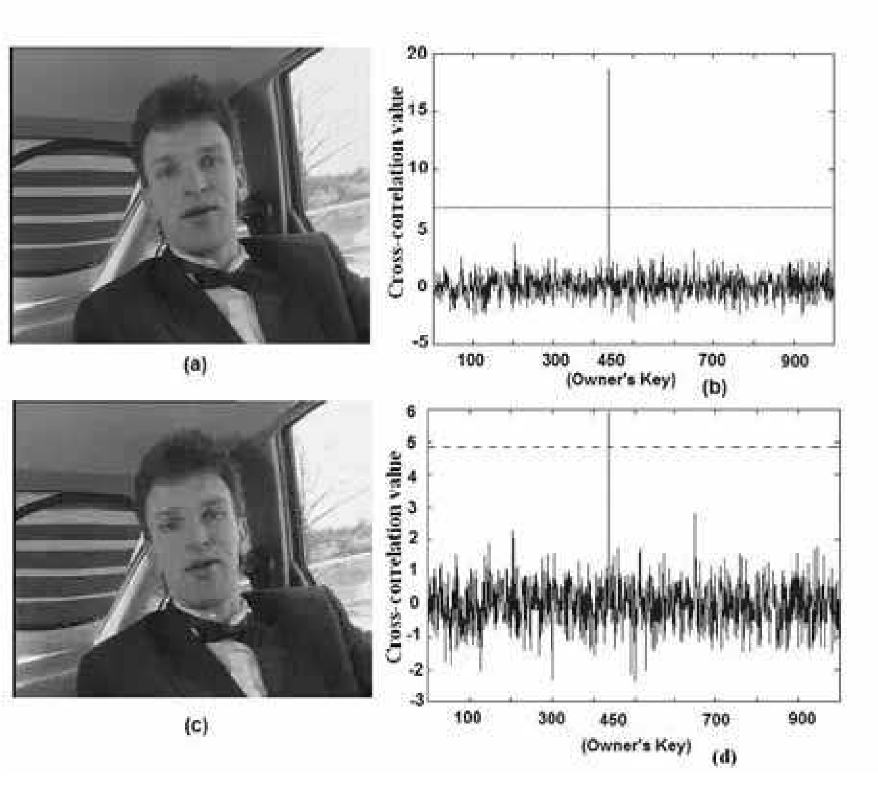

Fig. 9. (a) Watermarked frame, (c) watermarked and compressed frame, (b) and (d) detector responses for (a) and (c), respectively

The figure 10 shows the watermark robustness against impulsive noise contamination. The fig. 10(a) and fig. 10(c) show the watermarked frames that received impulsive noise contamination with a density of 3% and 10%, respectively; here fig. 10(b) and fig 10(d) are the detector responses of fig. 10(a) and fig. 10(c). Fig. 11 shows the watermark robustness against Gaussian noise contamination, here fig. 11(a) and fig. 11(c) show a watermarked and contaminated frame by Gaussian noise with variance 0.01 and 0.05, respectively; and fig. 11(b) and fig. 11(d) are detector responses of both cases, respectively. From these figures, we can conclude that the embedded watermark is sufficiently robust to impulsive and Gaussian noise contamination.

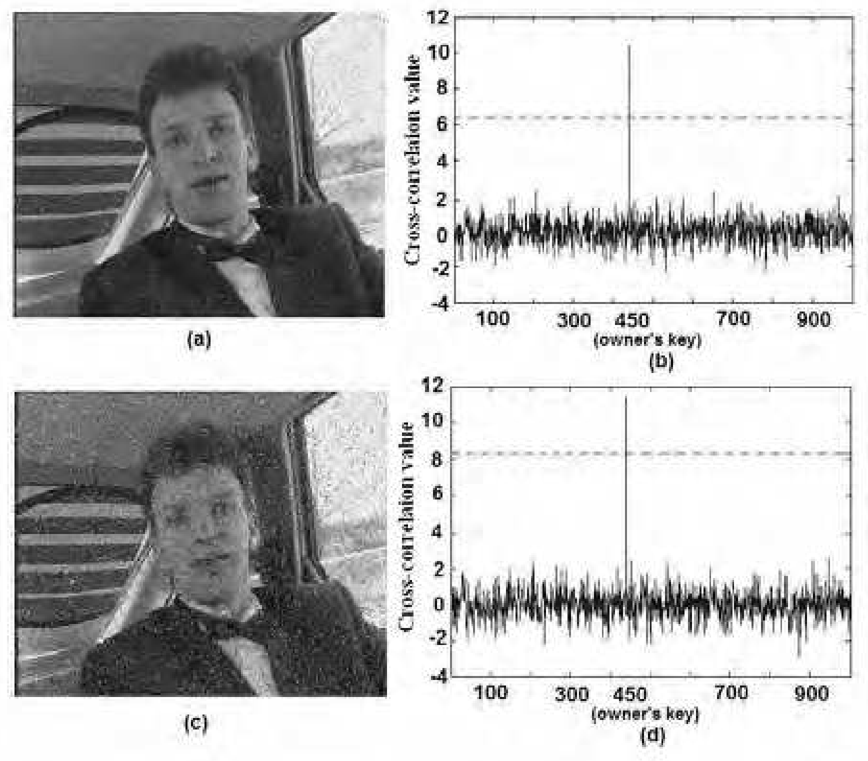

Fig. 10. Watermark Robustness against impulsive noise contamination.

Also in the proposed algorithm, the embedded watermark is sufficiently robust against cropping attack. The figure 12 shows the watermarked cropped frames and watermark detection performance from the cropped video sequence. Here fig. 12(a) and fig. 12(c) show a video sequence in which 40% and 75% of all frames of watermarked video sequence are cropped, and fig. 12(b) and fig. 12(d) are detector responses of both cases, respectively.

Frame dropping, frame swapping and frame averaging are intentional attacks for watermarked video sequences (Zhyang et al., 2004). These frame attacks take advantage of the temporal redundancy of video sequences and try to destroy efficiently the embedded watermark signal, without causing any visual degradation in the video sequence. Frame dropping, frame swapping and frame averaging attacks are described by (9), (10) and (11), respectively.

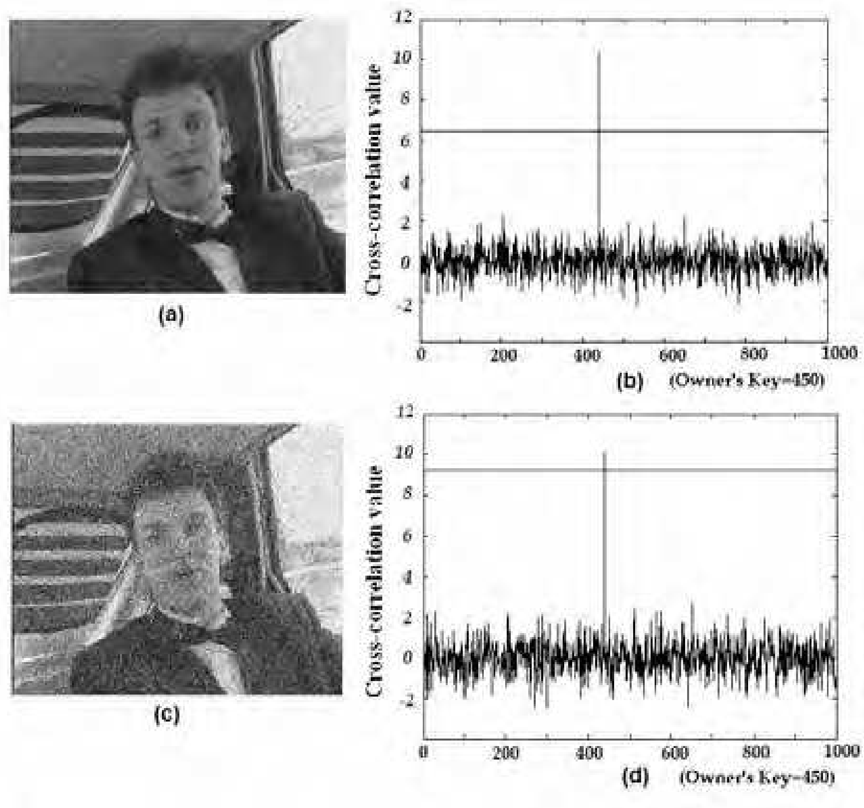

Fig. 11. Watermark Robustness against Gaussian noise contamination

$$V_{attacked} = V_{watermarked} - \{F_{r1}, F_{r2}, ...F_{rn}\} \tag{9}$$

where $V_{attacked}$ and $V_{watermarked}$ are attacked watermarked sequences by frame dropping and the watermarked sequence without attack, respectively, and $[r1, r2, ...rn]$ are the numbers of frames selected randomly.

$$\tilde{F}_k \Leftrightarrow F_{k+1} \quad \tilde{F}_{k+1} \Leftrightarrow F_k \tag{10}$$

$$\tilde{F}_k = \frac{1}{3}[F_{k-1} + F_k + F_{k+1}] \tag{11}$$

where $F_k, \tilde{F}_k$ are *k-th* frames of watermarked and attacked videos, respectively.

Because the proposed algorithm embeds watermark signals through temporal video sequences, the embedded watermark signal is inherently robust to frame attacks. Figure 13 shows the watermark robustness against frame averaging attack. Fig. 13(a) shows a result of averaging of one I-frame and two P-frames, and fig. 13(b) shows a result of averaging three P-frames; and fig. 13(b) and fig. 13(d) are the detector responses of both cases.

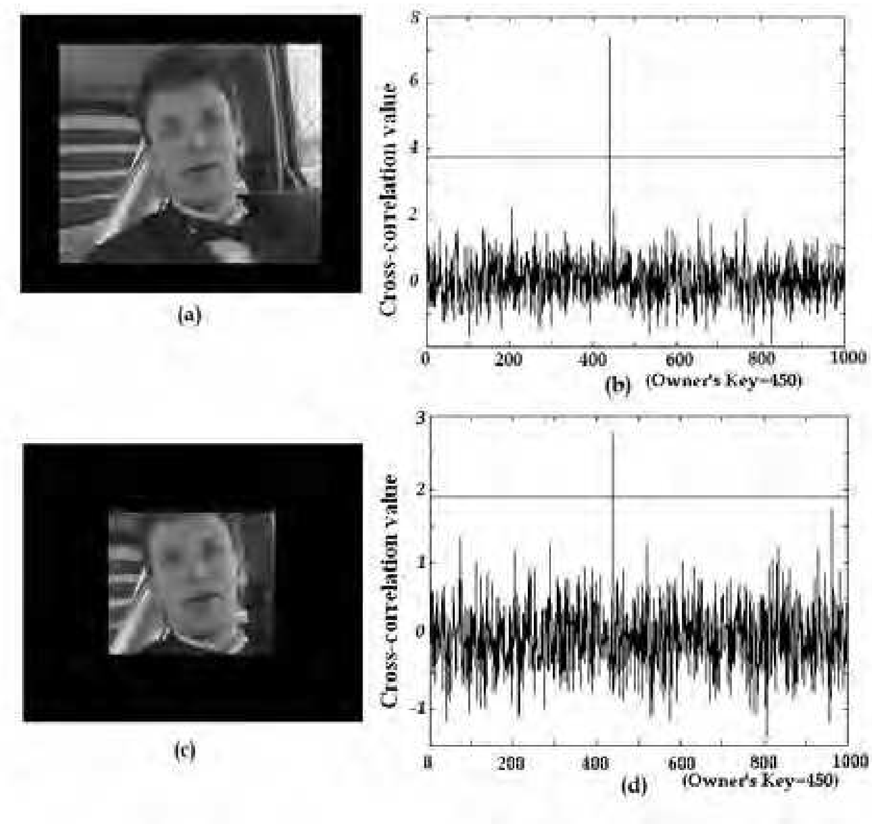

Fig. 12. Watermark robustness against cropping. (a) 40% of video data is cropped and (b) 75% of video data is cropped and (c) and (d) Detector responses of both cases.

3.3 Computational cost for watermark embedding and detection

In the video watermarking techniques, time consuming caused by watermark embedding and detection processes must be minimized. In our experiment, to evaluate the consumed time for watermarking operation, consider the processing time of MPEG coding with/ without the proposed watermarking process. As shown by table 2, the processing time with the watermarking operation increases approximately 40% for I-frames and 10% for P-frames, compared with the processing time without watermarking process. Considering that the number of P-frames is approximately 4 times larger than that of the I-frames and that the B-frames are excluded for watermarking process, the overall time required for watermarking operation is smaller than 10% of the total time required by the MPEG compression. This result means that the proposed watermarking algorithm is suitable for an actual implementation.

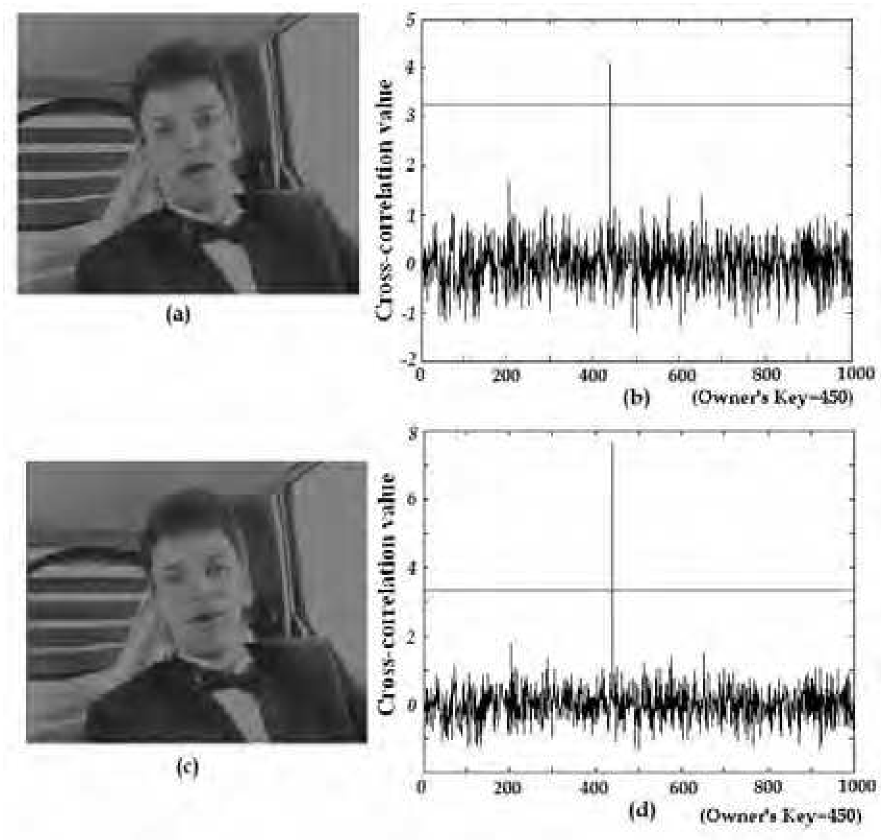

Fig. 13. (a) Frame averaging with I-frame, P-frames (b) detector response of (a), (c) Frame averaging with P-frames and (d) detector response of (c).

Frame Type	**Coding Time without watermarking**	**Coding time With watermarking**
I	0.83 sec / frame	1.2 sec / frame
P	0.3 sec / frame	0.33 sec / frame

Table 2. Computational cost of watermarking process

3.4 Comparison with other algorithms

The proposed algorithm is compared with other previously proposed algorithms with similar objective. Selected algorithms are as follows: A) Base band algorithm using 3D Wavelet Transform (Zhuang et al., 2004), B) The algorithm based on motion vector modification during MPEG coding (Zhao et al., 2003) and C) The watermarking algorithm based on the DCT domain, which performs during MPEG coding process (Zhang et al., 2001). The table 3 shows the watermark robustness comparison, here symbol 'O' means that the embedded watermark is robust against the indicated attack, while the symbol 'X' means that the embedded watermark cannot be detected after the attack is applied. As shown in

Attacks	A	B	C	Proposed
MPEG	**X**	**O**	**O**	**O**
Frame average	**O**	**O**	**X**	**O**
Frame dropping	**O**	**O**	**O**	**O**
Frame swapping	**O**	**O**	**O**	**O**
Cropping	**X**	**X**	**O**	**O**
Impulsive noise	**O**	**X**	**O**	**O**
Gaussian noise	**O**	**X**	**O**	**O**

Table 3. Comparison among three algorithms reported in literature and the proposed one.

4. Conclusions

The proposed algorithm performs watermark embedding and detection during MPEG-2 coding process, in which firstly an adequate embedding energy is computed using the HVS based three criteria: sensibility of different color channels (red, green and blue channel), sensibility of spatial region with different features, such as plain, texture and edge regions, and finally sensibility of regions with different motion speed. Due to the lower sensibility of the HVS to blue channel, the watermark embedding is carried out only in blue channel. And using the latter two criteria, an adequate watermark embedding energy is assigned to each block of 8×8 DCT coefficients of I-frames and P-frames. In the proposed algorithm, B-frames are not used for watermarking in order to reduce watermark embedding and detection time. The proposed algorithm was evaluated from watermark imperceptibility and robustness points of view. The watermark imperceptibility was evaluated using the PSNR and Universal Quality Index (UQI). Both values show the watermark imperceptibility of the proposed algorithm, especially perceptual distortion evaluated using UQI shows that the watermark is imperceptible by the HVS. To evaluate the watermark robustness against some common attacks including video frame attacks such as: frame dropping, frame swapping and frame averaging. The simulation results show the watermark high robustness to above mentioned attacks. Also the proposed algorithm is compared with other algorithms with similar objective; the comparison results show that the proposed watermarking algorithm is more robust against a wider range of attacks than other watermarking algorithms.

The additional processing time to the MPEG-2 standard coding caused by watermarking is also measured. Since in the proposed algorithm, watermarking is carried out only in the I-frames and P-frames, the overall additional time is less than 10% of the MPEG-2 standard coding. Therefore we can conclude that the proposed watermarking algorithm is suitable for a real implementation.

5. References

Biswas S., S. R. Das and E. M. Petriu, An Adaptive Compressed MPEG-2 Video Watermarking Scheme, IEEE Trans. on Instrumentation and Measurement: 54(5), 1853-1861 (2005).

Hartung, F. and B. Girod, Watermarking of uncompressed and compressed video, Signal Processing: 66, 283-301 (1998)

Kong W., B. Yang, D. Wu and X. Niu, SVD Based Blind Video Watermarking Algorithm, IEEE Int. Conf. on Innovative Computing, Information and Control: (2006)

Langelaar, G. C. and R. L. Lagendijk, Optimal differential energy watermarking of DCT encoded images and video, IEEE Trans. on Image Processing: 10(1), 148-158 (2001).

Liu, Z., H. Liang, X. Niu and Y. Yang, A Robust Video Watermarking in Motion Vectors, IEEE Int. Conf. Signal Processing: 2358-2361 (2004).

Noorkami M. and R. M. Mersereau, Improving Perceptual Quality in Video Watermarking Using Motion Estimation, IEEE Int. Conf. on Image Processing (ICIP): 2, 520-523 (2006).

Piva, A., M. Barni, F. Bartolini and V. Cappellini, DCT-Based Watermark Recovering without Resorting to the Uncorrupted Original Image, IEEE Int. Conf. on Image Processing (ICIP): 1, 520-523 (1997).

Plataniotis, N. and A. N. Venetsanopoulos, Color Image Processing and Application, First Edition, Springer-Verlag, (2000).

Sayood K., Introduction to Data Compression, 2nd Edition, Morgan Kaufmann Publishers (2000).

Swanson, M.D., B. Zhu and A. H. Tewfik, Multi-resolution scene-based video watermarking using perceptual models, IEEE J. Select areas Communication: 16, 540-550 (1998).

Tong, H. Y. and A. N. Venetsanopoulos, A Perceptual model for JPEG applications based on block classification, texture masking and luminance masking, Int. Conf. on Image Processing (ICIP), 3, 428-432 (1998).

Ueno, Y. and A Digital Video watermarking method by association with the Motion Estimation, IEEE Int. Conf. on Signal Processing (ICSP), 2576-2579 (2004).

Wang, J., A. R. Steele and J. Liu, Efficient Integration of Watermarking with MPEG Compression, IEEE Int. Conf. on Multimedia and Expo (ICME), 911-914 (2004)

Wang, Z., Lu L. and Bovik A. C., Video quality assessment based on structural distortion measurement, Elsevier Signal Processing: Image Communication: 19, 121-132 (2004).

Wolfgang, R. B, C. I. Podilchuk and E. J. Delp, Perceptual watermarks for digital images and video, Proceeding IEEE: 87(7): 1108-1126 (1999).

Zhao, Z., N. Yu and X. Li, A Novel Video Watermarking Scheme in Compressed Domain Based of Fast Motion Estimation, IEEE Int. Conf. on Communication Technology (ICCT), 1878-1882 (2003)

Zhang, J, J. Li, and L. Zhang, Video Watermark Technique in Motion Vector, XIV Brazilian Symposium on Computer Graphics and Image Processing (2001).

Zhuang, H., Y. Li and C. Wu, A Blind Spatial-temporal Algorithm based on 3D Wavelet for Video Watermarking, IEEE int. Conf. on Multimedia and Expo (ICME), 3, 1727-1730 (2004).

Visual Servoing

Edited by Rong-Fong Fung

ISBN 978-953-307-095-7

Hard cover, 234 pages

Publisher InTech

Published online 01, April, 2010

Published in print edition April, 2010

The goal of this book is to introduce the visional application by excellent researchers in the world currently and offer the knowledge that can also be applied to another field widely. This book collects the main studies about machine vision currently in the world, and has a powerful persuasion in the applications employed in the machine vision. The contents, which demonstrate that the machine vision theory, are realized in different field. For the beginner, it is easy to understand the development in the vision servoing. For engineer, professor and researcher, they can study and learn the chapters, and then employ another application method.