



Improving the robustness of DCT-based image watermarking against JPEG compression

Shinfeng D. Lin^a, Shih-Chieh Shie^{b,*}, J.Y. Guo^a

^a Department of Computer Science and Information Engineering, National Dong Hwa University, Hualien County, Taiwan, ROC

^b Department of Computer Science and Information Engineering, National Formosa University, 64 Wen-Hua Road, Hu-Wei, Yunlin County 632, Taiwan, ROC

ARTICLE INFO

Article history:

Received 1 February 2007

Received in revised form 4 April 2009

Accepted 23 June 2009

Available online 3 July 2009

Keywords:

Image watermarking

JPEG compression

Robustness

Torus automorphism

ABSTRACT

A DCT-based image watermarking technique is proposed in this article. To improve the robustness of watermark against JPEG compression, the most recently proposed techniques embed watermark into the low-frequency components of the image. However, these components hold significant information of the image. Directly replacing the low-frequency components with watermark may introduce undesirable degradation to image quality. To preserve acceptable visual quality for watermarked images, we propose a watermarking technique that adjusts the DCT low-frequency coefficients by the concept of mathematical remainder. Simulation results demonstrate that the embedded watermarks can be almost fully extracted from the JPEG-compressed images with very high compression ratios.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Along with the progress relating to computer hardware and software, the Internet has become the most popular channel for transmitting various forms of digital media. Since the environment of the Internet is open, the protection of digital images transmitted on the network has become an important research topic in recent years. Digital watermarking is a common technique to achieve copyright protection. For digital images, watermarking is the process of embedding significant data (watermark) into an image such that the embedded watermark can be detected or extracted later to make an assertion about the image [1,2].

Generally, a watermarking scheme consists of three parts, the watermark, the watermark embedding stage and the watermark verification stage. The watermark embedding algorithm incorporates the watermark into the host image, whereas the verification algorithm extracts and authenticates the watermark determining the ownership of the image. Usually, the watermark is a visually recognizable logo or a set of meaningless character strings that represents the copyright of the owner or legal users. If a watermark can be extracted from an image in the verification stage, it could prove the copyright of the owner [3].

In the past few years, two kinds of techniques that are related to but different from image watermarking have been introduced in

literature. One is image authentication technique and the other is image data hiding. The goal of image authentication is to verify the originality of an image by detecting malicious manipulations. However, most of the earlier methods for image authentication deal with all types of manipulations equally and unacceptably. That is these methods treat some practical manipulations such as image compression and image enhancement as attacks. Therefore, Lin and Chang proposed a DCT-based image authentication method which can prevent malicious manipulations but allow JPEG lossy compression [4]. In Lin and Chang's method, the authentication signatures for images are generated based on the invariance of the relationships between DCT coefficients at the same position in separate blocks of an image. More detailed theoretical analysis and experimental results can be found in [4]. The purpose of image data hiding is different from traditional cryptography. Cryptography concentrates on encrypting meaningful messages into meaningless data while image data hiding covers secret information with the host image as camouflage. Hiding data in images involves embedding a large amount of secret data into a cover image with minimal perceptible degradation of image quality. However, the hiding capacity for secret data and the distortion of the cover image are a tradeoff since more hidden data always result in more degradation on the cover image.

Digital watermarking has been shown as a valid solution to the problem of image copyright protection [5,6]. There are two essential requirements for image watermarking. One is invisibility, namely the watermarked image should not be perceived the changes of the original image by human eyes. It means that the visual quality of watermarked images should not be destroyed by the embedding of

* Corresponding author.

E-mail address: scshie@nfu.edu.tw (S.-C. Shie).

watermark. The other is robustness, namely the watermark should be able to resist attacks, even if these attacks are deliberately made [7]. To improve the robustness of watermark, most of the recently proposed techniques embed watermark information into the low-frequency part of images [8–10]. Huang et al. proposed a novel DCT-based technique which embeds watermarks in the DC components of images [8]. The main idea behind this technique is that more robustness can be achieved if watermarks are embedded in DC components since DC components have much larger perceptual capacity than ac components. In addition, the authors took advantages of the feature of spatial masking (both luminance and texture masking) of the HVS to adaptively embed watermarks into images. For digitized images to be safely and efficiently transmitted on the Internet, watermarked images should be particularly robust to JPEG compression. Recently, Lin et al. proposed two kinds of DCT-based image watermarking techniques [9,10]. These schemes perform well under general JPEG compression. However, when the watermarked images have to be compressed with higher compression ratio, the embedded watermarks may be destroyed seriously. To overcome this problem, an improved technique has been studied and proposed in this paper.

The rest of this paper is organized as follows. The proposed DCT-based watermark embedding and extracting algorithms are described in Sections 2 and 3, respectively. The relative parameters for the proposed watermarking scheme are provided in Section 4. Section 5 addresses our experimental results and discussion, especially on JPEG compression with very high compression ratios. Finally, the conclusions are given in Section 6.

2. Watermark embedding algorithm

The detailed steps of the proposed watermark embedding algorithm are given as follows.

Step 1. Torus automorphism permutation: in order to increase the security and robustness of watermark, the watermark pattern should be disarranged before embedding. Torus automorphism (TA) is an effective method to disperse a watermark equally and randomly [11]. Applying the concept of TA for scrambling the binary watermark before it is embedded into the host image offers cryptographic protection against intentional reconstruction of watermark [12]. This is because the key utilized in the TA permutation procedure (for scrambling the watermark) is also necessary in the inverse TA permutation procedure (for reconstructing the watermark). The key is held only by the distributor of the watermarked image. Without the key, the attacker cannot determine the original permutation of watermark bits. Consequently, it is difficult for the attacker to detect and reconstruct the watermark pattern. The details of the Torus automorphism permutation can be found in [13].

The watermark used in the proposed scheme is permuted based on the following equation before it is embedded into the host image.

$$\begin{pmatrix} i^* \\ j^* \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ k & k+1 \end{pmatrix} \times \begin{pmatrix} i \\ j \end{pmatrix} \bmod m \quad (1)$$

Eq. (1) means that each pixel of the watermark pattern at coordinate (i, j) is moved to (i^*, j^*) , and k and m are the key parameters given by the user. Let W_p be the permuted watermark after Torus automorphism permutation, then divide W_p into non-overlapping pairs.

Step 2. YUV color transformation: RGB color space is highly correlated and is not suitable for watermarking applications, except for the blue channel utilized by some researchers. However, the potential of the RGB channels can be exploited for watermarking, by decreasing the correlation among them. In the proposed scheme, we adopt the YUV color space for watermark embedding. The watermark information is embedded into the luminance components of images. The luminance information Y of the color image is obtained by

applying the YUV color transformation, as shown in Eq. (2), on the original RGB image. The two reasons for embedding watermark in Y (luminance) rather than U or V (chrominance) are: (i) human visual system is more sensitive to luminance than to the other two chrominance components, and (ii) the JPEG and MPEG standards typically use higher density for Y than for the other two components.

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.148 & -0.289 & 0.437 \\ 0.615 & -0.515 & -0.100 \end{pmatrix} \times \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (2)$$

Step 3. DCT transformation and quantization: Discrete Cosine Transform (DCT) is a kind of signal decomposition that converts images from spatial domain to frequency domain. The equation of forward DCT regarding an image block of size $N \times N$ pixels is given in Eq. (3). Note that $DCT(i, j)$ represents the coefficient at coordinate (i, j) in the DCT-transformed block and $pixel(x, y)$ is the pixel value at coordinate (x, y) in the original block.

$$DCT(i, j) = C(i) \times C(j) \times \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} pixel(x, y) \times \cos\left[\frac{(2x+1)i\pi}{2N}\right] \cos\left[\frac{(2y+1)j\pi}{2N}\right] \quad (3)$$

where

$$C(i), C(j) = \begin{cases} \sqrt{\frac{1}{N}} & \text{for } i, j = 0 \\ \sqrt{\frac{2}{N}} & \text{for } i, j = 1, 2, \dots, N-1 \end{cases}$$

After color transformation, the luminance Y of image is divided into non-overlapping blocks of size 8×8 , and each block of Y is DCT transformed independently. In order to enhance the robustness of watermark against JPEG compression, each DCT-transformed block of Y is quantized based on the quantization table provided by the JPEG compression standard as shown in Fig. 1.

Step 4. Embedded blocks selection: to improve the invisibility of watermark, the complex blocks of images are selected for watermark embedding. In the DCT domain, the number of non-zero coefficients in each block is computed to estimate the complexity of the image block. Let $N(n)$ represent the number of non-zero coefficients in block n . The image blocks can be sorted into descending order based on their corresponding $N(n)$ s.

Step 5. Watermark insertion: in the proposed scheme, the binary information of watermark pattern W is sequentially embedded into the selected image blocks obtained in step 4. Two bits of watermark pattern are embedded into the low-frequency components of each selected block for preserving the visual quality of watermarked

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Fig. 1. The quantization table recommended in the JPEG specification.

images and increasing the robustness of watermark. Here, the low-frequency components of image block can be the DCT coefficients at positions $C(2, 0)$, $C(1, 1)$, $C(0, 2)$, $C(0, 3)$, $C(1, 2)$, $C(2, 1)$, and $C(3, 0)$. The positions of DCT coefficients within an image block are shown as Fig. 2.

To embed watermark information, we have to modify the DCT coefficients at the low-frequency positions. Let C be the value of DCT coefficient and M be the modulus, the other relative variables can be defined as the following formulas

$$\begin{aligned} r &= |C| \bmod M \\ q &= \frac{|C|}{M} \\ \text{sign} &= \begin{cases} 1 & , \text{if } C \geq 0 \\ -1 & , \text{if } C < 0 \end{cases} \end{aligned} \quad (4)$$

In Eq. (4), r is the mathematical remainder of $|C|$ and $r \in \{0, 1, 2, \dots, M-1\}$, q is the mathematical quotient obtained by dividing $|C|$ by M , and sign represents that C is positive or negative.

The binary information of watermark pattern is embedded into the selected block by changing the value of coefficient C . Let C^* be the modified coefficient corresponding to C . The value of C^* is obtained by the following rule.

If watermark bit = 0:

$$\begin{aligned} r' &= \frac{M}{4} \\ C_{\text{low}} &= \text{sign} \times (q \times M + r') \\ C_{\text{high}} &= \text{sign} \times ((q + 1) \times M + r') \\ C^* &= \begin{cases} C_{\text{low}} & , \text{if } |C_{\text{low}} - C| \leq |C_{\text{high}} - C| \\ C_{\text{high}} & , \text{if } |C_{\text{low}} - C| > |C_{\text{high}} - C| \end{cases} \end{aligned}$$

If watermark bit = 1:

$$\begin{aligned} r' &= \frac{3M}{4} \\ C_{\text{low}} &= \text{sign} \times ((q - 1) \times M + r') \\ C_{\text{high}} &= \text{sign} \times (q \times M + r') \\ C^* &= \begin{cases} C_{\text{low}} & , \text{if } |C_{\text{low}} - C| \leq |C_{\text{high}} - C| \\ C_{\text{high}} & , \text{if } |C_{\text{low}} - C| > |C_{\text{high}} - C| \end{cases} \end{aligned}$$

Note that C_{low} and C_{high} are the two candidate values adaptively generated for replacing C . Based on the above rule, C^* can be easily computed and the difference between C^* and C is very small. In the

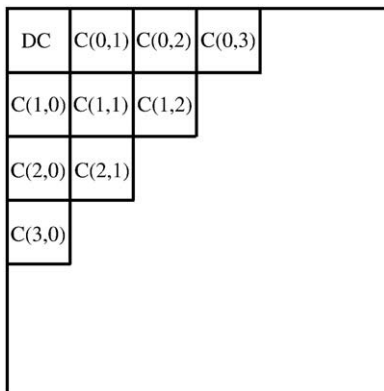


Fig. 2. The positions of DCT coefficients within an image block.



Fig. 3. Watermark pattern used in the experiments.

worst condition, the largest difference between C and C^* is $M/2$. Cooperated with the watermark extracting algorithm, the proposed watermark embedding rule provides a safe range for the value of C^* . This means that even the value of C^* is changed into a new value C^{**} with a difference as large as M (due to image processing attacks) the embedded watermark bit can be still extracted successfully.

Step 6. Inverse DCT transformation: after all the watermark bits are inserted into the selected image blocks, each block of the host image is inverse DCT transformed independently. The equation of inverse DCT is given in Eq. (5).

$$\begin{aligned} \text{pixel}(x, y) &= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} C(i) \times C(j) \times \text{DCT}(i, j) \\ &\times \cos\left[\frac{(2x+1)i\pi}{2N}\right] \cos\left[\frac{(2y+1)j\pi}{2N}\right] \end{aligned} \quad (5)$$

where

$$C(i), C(j) = \begin{cases} \sqrt{\frac{1}{N}} & \text{for } i, j = 0 \\ \sqrt{\frac{2}{N}} & \text{for } i, j = 1, 2, \dots, N-1 \end{cases}$$

Step 7. Weighted correction: to further improve the visual quality of the watermarked image, the weighted correction procedure is applied and it makes a slight change on the watermarked (modified) luminance Y^* . Let D be the difference between the original luminance Y and the watermarked luminance Y^* , the weighted correction procedure can be represented by Eq. (6).

$$\begin{aligned} D &= Y - Y^* \\ Y^{**} &= Y^* + f \times D \end{aligned} \quad (6)$$

In Eq. (6), Y^{**} is the final luminance of the watermarked image and f is a quality factor with a value between 0 and 1. When f is with a larger value, Y^{**} is more close to Y and the visual quality of the final

Table 1

NC values under JPEG compression with different compression ratios (CR: compression ratio; PSNR: in the unit of dB).

M	CR	0	10	20	30	40	43	47	53
14	PSNR	36.4	35.0	34.4	32.3	31.0	30.4	29.9	29.3
	NC	1	1	0.97	0.96	0.92	0.44	0.46	0.44
18	PSNR	36.0	34.7	34.1	32.2	30.7	30.4	29.8	29.1
	NC	1	1	1	0.94	0.95	0.72	0.37	0.43
26	PSNR	34.4	33.5	33.0	31.4	30.5	30.1	29.4	28.7
	NC	0.97	0.97	0.97	0.97	0.93	0.96	0.97	0.97

Table 2

NC values under uniform noise attacks (NR: noise ratio in %).

M	NR	0	5	10	15	20	25	30
14	PSNR	36.4	29.8	24.6	21.2	18.7	16.9	15.4
	NC	1	0.99	0.72	0.48	0.31	0.23	0.12
18	PSNR	36.0	29.6	24.5	21.1	18.7	16.9	15.4
	NC	1	1	0.90	0.68	0.45	0.32	0.24
26	PSNR	34.4	29.0	24.3	21.1	18.7	16.8	15.4
	NC	0.97	0.97	0.96	0.87	0.71	0.56	0.42

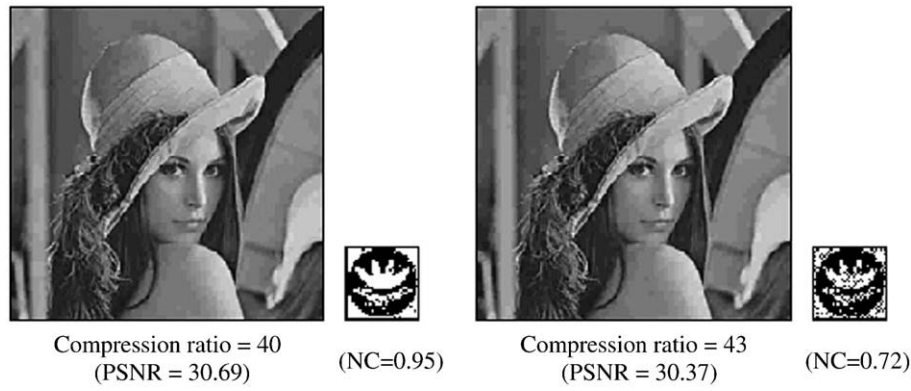


Fig. 4. Watermarks retrieved from marked images *Lena* after JPEG compression with compression ratios 40 and 43, respectively, when the modulus M is 18.

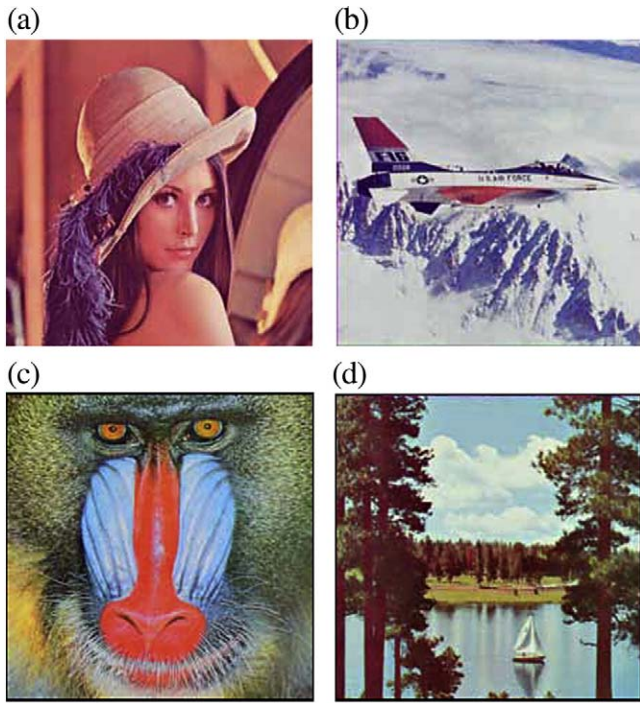


Fig. 5. The watermarked color images. (a) *Lena*, PSNR = 31.47, (b) *F16*, PSNR = 33.9, (c) *Baboon*, PSNR = 26.94, and (d) *Scene*, PSNR = 30.53.

watermarked image is better. However, the larger value of f will affect the watermark extraction rate. Therefore, the value of f should not be too large.

Step 8. Inverse YUV color transformation: The final step of the proposed watermark embedding algorithm is to transform the YUV color planes of the watermarked image back to the original RGB color planes. Instead of the watermarked luminance Y^* , the final watermarked image is acquired by using the luminance Y^{**} obtained in step 7.

3. Watermark extracting algorithm

The steps of the proposed watermark extracting algorithm are very similar to that of watermark embedding algorithm, except for the step of watermark information extraction. The procedure for watermark extraction is quite simple and it doesn't need any assistance of the original host image. Note that two key parameters are needed in the watermark extraction procedure: the record of selected image blocks that were embedded with watermark bits and the modulus M . In addition, the parameters of Torus permutation function are also needed when reconstructing the watermark pattern. The steps for watermark extraction are briefly listed as follows.

Step 1. Extract the luminance information of watermarked image based on the YUV color transformation.

Step 2. Divide the luminance information of watermarked image into non-overlapping blocks of size 8×8 . Transform each block into DCT frequency domain independently based on the DCT transformation.

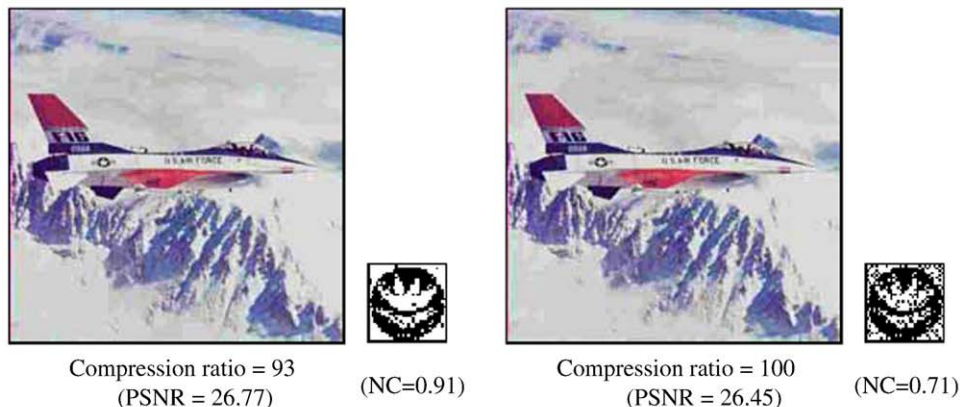


Fig. 6. Watermarks retrieved from marked images *F16* after JPEG compression with compression ratios 93 and 100, respectively, when the modulus M is 18.

Table 3

NC values under JPEG compression with different quality factors (CR: compression ratio).

Quality factor		90	60	30	20	15	10	9	8
<i>Lena</i>	CR	15.6	26.2	46.8	63.4	78.2	102.5	111.7	124.0
	PSNR	29.5	28.8	28.5	27.8	27.2	26.1	25.8	25.4
	NC	1	0.99	0.98	0.98	0.90	0.96	0.73	0.49
<i>F16</i>	CR	15.5	25.0	43.3	57.7	70.4	93.3	100.7	110.3
	PSNR	31.5	30.3	29.8	28.8	28.0	26.8	26.5	26.1
	NC	1	1	1	1	0.84	0.91	0.71	0.40
<i>Baboon</i>	CR	7.3	13.1	22.3	31.0	39.1	55.6	61.9	69.1
	PSNR	24.7	24.2	23.6	22.9	22.4	21.5	21.3	21.0
	NC	1	1	1	1	0.84	0.88	0.54	0.47
<i>Scene</i>	CR	11.0	18.8	32.5	44.1	54.8	74.5	81.2	89.5
	PSNR	27.7	26.8	26.4	25.7	25.2	24.1	23.9	23.6
	NC	1	0.99	0.99	0.96	0.85	0.85	0.57	0.46

Step 3. Extract the binary information of watermark based on the following rule. In Eq. (7), b is the extracted watermark bit and $C(i, j)$ is the DCT coefficients at the low-frequency positions where the watermark bits are embedded.

$$b = \begin{cases} 0, & \text{if } (|C(i, j)| \bmod M) < \frac{M}{2} \\ 1, & \text{if } (|C(i, j)| \bmod M) \geq \frac{M}{2} \end{cases} \quad (7)$$

Step 4. Rearrange the extracted watermark bits into right order by Torus automorphism permutation. Then the watermark pattern is reconstructed.

4. Secret keys

Secret keys are necessary in the proposed watermark embedding and extracting algorithms. Four kinds of secret keys are utilized in the proposed scheme. These secret keys should be preserved well for watermark extraction and verification.

























1. The parameters for Torus automorphism permutation: Torus automorphism permutation has been introduced in Section 2. The disarranging parameters k and m were defined in the first step of watermark embedding algorithm. These parameters are necessary when reconstructing the extracted watermark pattern.
2. The record of selected image blocks: the record of image blocks selected for watermark bits embedding has to be preserved well for watermark information extraction.
3. The embedding positions of watermark bits: the number of DCT coefficients and the positions of them that were designed for watermark bits embedding have to be protected well for watermark extraction.
4. The modulus used in the proposed algorithm: the modulus M used in the watermark information embedding step is necessary in the step of watermark bits extracting.

5. Simulation results and discussion

The proposed scheme has been conducted on gray-level and color images, respectively. All these test images are with size 512×512 pixels. The watermark pattern used in the experiments is a binary image with size 32×32 pixels and it is given as Fig. 3. To provide objective judgment of the extracting fidelity, the similarity measurement, the Normalized Correlation (NC) value between the original watermark W and the extracted watermark W^* , is applied and it is defined as Eq. (8). In addition, the visual quality of watermarked

Table 4

Simulation results of common image processing attacks.

Attacks	Auto levels	Auto contrast	Rotation(3°cw)	Diffuse glow
Attacked images				
PSNR (dB)	20.04	27.53	30.17	13.59
Extracted watermarks				
NC	0.775	0.998	0.830	0.537
Attacks	Despeckle	Cropping	Sharpen	Sharpen edge
Attacked images				
PSNR (dB)	30.51	24.22	28.00	30.32
Extracted watermarks				
NC	0.797	0.824	0.969	1
Attacks	Unsharp mask	High pass	Invert	Equalize
Attacked images				
PSNR (dB)	29.72	14.47	6.94	22.59
Extracted watermarks				
NC	0.982	1	1	0.854

images is evaluated by the peak signal-to-noise ratio (PSNR) criterion defined as Eq. (9).

$$NC = \frac{\sum_{i=0}^{31} \sum_{j=0}^{31} W(i, j) \times W^*(i, j)}{\sum_{i=0}^{31} \sum_{j=0}^{31} [W(i, j)]^2} \quad (8)$$

$$PSNR = 10 \log_{10} \frac{E_{\max}^2 \times I_w \times I_h}{\sum (I_{x,y} - I_{x,y}^*)^2} \quad (9)$$

In Eq. (9), I_w and I_h are the width and height of the watermarked image, respectively. $I_{x,y}$ is the original image pixel value at coordinate

Table 5

Performance comparison among [9,10], and proposed scheme in terms Of JPEG compression (CR: compression ratio).

	CR	10	20	30	40	50	53
[9]	PSNR	35.4	34.8	NA	NA	NA	NA
	NC	1	0.29	NA	NA	NA	NA
[10]	PSNR	35.4	34.8	32.6	31.1	NA	NA
	NC	0.99	0.50	0.12	0.07	NA	NA
Proposed	PSNR	33.5	33.0	31.4	30.5	29.0	28.7
	NC	0.97	0.97	0.97	0.93	0.96	0.97

(x, y) and $I_{x,y}^*$ is the altered image pixel value at coordinate (x, y) . E_{max} is the largest energy of the image pixels (i.e., $E_{max} = 255$ for 256 gray-level images). Moreover, the compression ratio of the JPEG-compressed image is obtained by dividing the file size of the original uncompressed image by that of the JPEG-compressed image.

The simulation results of the gray-level image, *Lena*, are given in Tables 1 and 2. Table 1 lists the NC values of extracted watermarks and the image quality (i.e., the PSNR values) of watermarked images after attacked by JPEG compression with different compression ratios. It shows the embedded watermark can be almost fully extracted from the JPEG-compressed image with compression ratio 53 when the modulus M is 26. Table 2 shows the NC values of extracted watermarks after attacked by adding different percentages of uniform noise on watermarked image. Note that in Tables 1 and 2, the NC values are not equal to 1 when the modulus M equals 26 in the circumstance that no attack has been performed on the watermarked image. This is because the steps of inverse DCT transformation, the weighted correction, and the inverse YUV color transformation in the proposed watermark embedding algorithm introduce a little distortion on the embedded watermark information. Fig. 4 shows the watermarks retrieved from the watermarked image *Lena* after attacked by JPEG compression with compression ratios 40 and 43, respectively.

The simulation results of the color images, *Lena*, *F16*, *Baboon*, and *Scene*, are given in Figs. 5 and 6 and Tables 3 and 4. The modulus M for watermark embedding is 18 in this experiment. Table 3 shows the NC values of extracted watermarks after attacked by JPEG compression with different quality factors. Note that the quality factor for images is an integer value ranging from 1 to 99, which denotes the predetermined image quality. The larger the quality factor is assigned, the lower compression ratio the compressed image obtains and the better visual quality the compressed image retains. Fig. 5 illustrates the watermarked color images, together with their PSNR values. It reveals that good visual quality of watermarked images can be obtained by the proposed scheme. Fig. 6 shows the watermarks retrieved from the marked image *F16* after JPEG compression with high compression ratios 93 and 100, respectively. As shown in Fig. 6, the extracted watermarks can be easily recognized by the human eyes. To show the robustness of the proposed watermarking scheme against common image processing attacks other than JPEG compression, we conduct the options provided in the software of PhotoShop on the marked image *Lena*. The marked images after being attacked and the retrieved watermarks are listed in Table 4. As shown in Table 4, the proposed scheme is robust to common image processing attacks such as cropping, rotation, sharpening, invert, and so on. These experimental results illustrate that a noticeable improvement, both on the robustness and on the imperceptibility, is achieved by the proposed scheme.

To show the improvement of the proposed scheme in terms of JPEG compression, we compare our scheme with the earlier works, [9] and [10], and the performance comparison is given in Table 5. Note that “NA” means the corresponding experimental datum is unavailable. As shown in Table 5, the watermark embedded by the proposed scheme can be almost fully extracted from the marked image when the compression ratio is 53. However, under the same circumstance, the watermarks embedded by [9] and [10] cannot survive after attacked by JPEG compression with compression ratios 20 and 25, respectively. It demonstrates that the proposed scheme achieves a good improvement on the robustness against JPEG compression. Note that in Tables 1, 3, and 5, some of the NC values increase when the compression ratio increases. This is because both the proposed watermarking system and the JPEG compression system are not linear systems. For the robustness at high compression ratio and the visual quality of marked image, the applied steps or parameters in the proposed algorithm may introduce a little distortion on the embedded watermark information. In addition, during JPEG compression attack, the marked image has to undergo several nonlinear transformation or quantization procedures. Therefore, a small number of the embedded watermark bits can be distorted at higher compression ratio.

Note that although there are several alternative DCT coefficients at the low-frequency positions, the watermark bits are embedded into the coefficients at positions $C(0, 2)$ and $C(2, 0)$ in our experiments. The reasons why we choose the two components to embed watermark are summarized as follows. The DCT coefficients at the positions $C(0, 1)$, $C(1, 0)$, and $C(1, 1)$ possess more significant information of image. Embedding watermark into these positions based on the proposed scheme introduces too much quality degradation on the marked image. In addition, it also makes the watermark information more perceptible. The DCT coefficients at the positions $C(0, 3)$, $C(1, 2)$, $C(2, 1)$, and $C(3, 0)$ are the higher-frequency components of image. Embedding watermark into these components by the proposed scheme makes it easy to detect the existence of watermark by statistic methods. In addition, to survive high-ratio JPEG lossy compression, it is improper to embed watermark in the higher-frequency components of images. Therefore, for the purpose of preserving good visual quality for marked images and increasing the robustness of watermark, the watermark information are embedded in the AC components at positions $C(0, 2)$ and $C(2, 0)$.

Table 6 shows the quantitative analysis of modulus M with respect to the visual quality of marked images and the robustness of embedded watermarks. We use sixteen gray-level and color images in this experiment. The averaged PSNR and NC values with respect to the modulus M are listed in this table. The experimental data demonstrate that the larger the modulus M is, the more robustness the watermark obtains. However, a larger modulus introduces more distortion on the

Table 6
Modulus value M with respect to visual quality of marked image and robustness of watermark (CR: compression ratio).

M	CR	0	10	20	30	40	50	60	70	80	90	100
4	PSNR	39.4	35.3	32.8	30.9	29.7	28.4	27.1	26.7	26.4	26.1	25.3
	NC	1	1	0.98	0.96	0.93	0.84	0.76	0.61	0.49	0.43	0.36
8	PSNR	37.7	34.9	32.6	30.4	29.1	27.9	27.0	26.5	26.3	26.1	25.1
	NC	1	1	0.98	0.96	0.94	0.87	0.81	0.71	0.65	0.57	0.42
12	PSNR	36.9	34.2	31.8	30.1	29.0	27.7	27.0	26.3	26.1	25.8	25.1
	NC	1	1	1	0.99	0.93	0.90	0.83	0.74	0.65	0.56	0.43
16	PSNR	35.3	33.3	31.5	30.1	28.7	28.0	27.4	26.3	26.0	25.6	25.1
	NC	1	1	0.99	0.99	0.98	0.93	0.91	0.84	0.79	0.66	0.57
20	PSNR	34.2	32.4	31.0	29.8	28.6	28.0	27.3	26.1	25.5	25.3	24.9
	NC	1	1	0.98	0.97	0.97	0.97	0.94	0.89	0.83	0.76	0.67
24	PSNR	32.4	32.0	31.0	29.6	28.5	27.4	26.9	25.8	25.2	24.9	23.4
	NC	1	0.99	0.98	0.98	0.98	0.96	0.97	0.89	0.85	0.79	0.70
28	PSNR	31.3	30.5	29.7	29.0	28.1	27.3	26.1	25.3	24.8	24.1	22.9
	NC	1	0.99	0.99	0.97	0.98	0.95	0.92	0.90	0.85	0.80	0.71

marked images. In our opinion, the modulus M should be set a value between 16 and 26.

6. Conclusions

An improved DCT-based image watermarking technique has been introduced in this paper. The watermark is embedded into a digital image, based on the concept of mathematical remainder, by modifying the low-frequency coefficients in DCT frequency domain. With the proposed scheme, the embedded watermark can successfully survive after attacked by image processing operations, especially for the JPEG compression with high compression ratio. Moreover, the watermark embedding and extracting processes are very simple and the watermark is self-extractable. Simulation results show that the proposed scheme outperforms the earlier works. Therefore, we conclude that the new proposed technique is more suitable for images that will be highly JPEG-compressed and transmitted on the Internet.

Acknowledgements

The authors are supported by National Science Council (Taipei, Taiwan, ROC), National Dong Hwa University (Hualien, Taiwan, ROC), and National Formosa University (Hu-Wei, Yunlin, Taiwan, ROC).

References

- [1] J.L. Liu, D.C. Lou, M.C. Chang, H.K. Tso, A robust watermarking scheme using self-reference image, *Computer Standards & Interfaces* 28 (2006) 356–367.
- [2] D.C. Lou, H.K. Tso, J.L. Liu, A copyright protection scheme for digital images using visual cryptography technique, *Computer Standards & Interfaces* 29 (2007) 125–131.
- [3] K. Gopalakrishnan, N. Memon, P.L. Vora, Protocols for watermark verification, *IEEE Transactions on Multimedia* 8 (4) (2001) 66–70.
- [4] C.Y. Lin, S.F. Chang, A robust image authentication method distinguishing JPEG compression from malicious manipulation, *IEEE Transactions on Circuits and Systems for Video Technology* 11 (2) (2001) 153–168.
- [5] W. Lu, H. Lu, F.L. Chung, Novel robust image watermarking using difference correlation detector, *Computer Standards & Interfaces* 29 (2007) 132–137.
- [6] M.D. Swanson, M. Kobayashi, A.H. Tewfik, Multimedia data-embedding and watermarking technologies, *Proceedings of the IEEE* 86 (6) (1998) 1064–1087.
- [7] C.D. Vleeschouwer, J.F. Delaigle, B. Macq, Invisibility and application functionalities in perceptual watermarking an overview, *Proceedings of the IEEE* 90 (1) (2002) 64–77.
- [8] J. Huang, Y.Q. Shi, Y. Shi, Embedding image watermarks in DC components, *IEEE Transactions on Circuits and Systems for Video Technology* 10 (6) (2000) 974–979.
- [9] S.D. Lin, C.F. Chen, A robust DCT-based watermarking for copyright protection, *IEEE Transactions on Consumer Electronics* 46 (3) (2000) 415–421.
- [10] S.D. Lin, S.C. Shie, C.F. Chen, A DCT based image watermarking with threshold embedding, *International Journal of Computers and Applications* 25 (2) (2003) 130–135.
- [11] C.C. Chang, J.Y. Hsiao, C.L. Chiang, An image copyright protection scheme based on torus automorphism, *Proceedings of the First International Symposium on Cyber Worlds* (2002) 217–224.
- [12] M. Engedy, V.N.K. Munaga, A. Saxena, A robust wavelet based digital watermarking scheme using chaotic mixing, *Proceedings of the First International Conference on Digital Information Management* (2006) 36–40.
- [13] G. Voyatzis, I. Pitas, Chaotic mixing of digital images and applications to watermarking, *Proceedings of the European Conference on Multimedia Applications, Services and Techniques* 2 (1996) 687–695.