An Image Copyright Protection Scheme Using DCT and a Table Look-up Technique

Chi-Nan Lin[1, 2], Chin-Chen Chang[1, 3], Mien-Tsung Tsai[4]
[1]Department of Computer Science and Information Engineering
National Chung Cheng University
Chiayi 62102, Taiwan, R.O.C.
lcn@cs.ccu.edu.tw

[2]Department of Management Information Systems
Central Taiwan University of Science and Technology
Taichung 40601, Taiwan, R.O.C.
cnlin@ctust.edu.tw

[3]Department of Information Engineering and Computer Science
Feng Chia University
Taichung 40724, Taiwan, R.O.C.
ccc@cs.ccu.edu.tw

[4]Department of Information Engineering and Computer Science
National Chi Nan University
Puli, Nantou Hsien 545, Taiwan, R.O.C.
mttsai.study@gmail.com

Journal of Digital Information Management

***ABSTRACT**: In this paper, an image copyright protection scheme was proposed which will not involve watermark embedment in the host image. The proposed scheme is suitable for protecting precious image work where the host image should not be altered. Multiple copies of unruly black and white binary images were created by the XORed entwinement with the binary copyright logo (i.e., the watermark) and the binary characteristic extraction of host image's DCT-coefficients. The multiple copies of unruly binary images were then time-stamped with a digital signature and registered with a certified authority for protecting the rightful ownership. A giveup-table was employed to contain possible attacked host image locations and a simple table look-up operation was applied to select better candidates from the multiple copies of unruly binary images for recovering the watermark. Experimental testing will be conducted on attacks such as blurring, rescale, cropping, sharpening, noise, rotation, and JPEG lossy compression. Results showed that the watermark for proof of rightful ownership can be recovered at a high NC > 0.95 and is visually undistorted.*

Categories and Subject Descriptors
K.5.1 [Legal aspects of computing]: Hardware/software protection—Copyrights; **I.4.9 [Image procesing and computer vision]:** Applications

General Terms
Legal aspects, security, verification

Keywords: Copyright protection, watermark, discrete cosine transform (DCT), table look-up

Received on 12 January 2009; Revised 17 May 2009; Accepted 30 June 2009

1. Introduction

As more and more applications moved on to the internet, data are easily accessible and vulnerable to piracies. The massive download, copy, reedit or redistribution of data without the permission of the owner, have incurred great losses to the copyright owners. Copyright protection for intellectual property on the Internet is indeed an important issue.

Techniques for hiding a digital watermark (i.e., the copyright logo) in the host image have been developed to protect its rightful ownership [2,3,4,8,10]. The watermark embedment can be in either the spatial or the frequency domain. Spatial domain embedment involved sophisticated manipulation of binary bits in the visual cryptography technique [6] or hiding in the least significant bits (LSB) of a pixel [3]. The frequency domain embedment could be discrete wavelet transformed [5,11,14] or discrete cosine transformed [1,7,9]. Normally, the watermark hidden in the spatial domain is fragile and not robust to image manipulations. Although hiding in the frequency domain is more robust and not easy to detect, the host image is usually modified from the replacement of the watermark bits.

This paper proposes an unorthodox image copyright protection scheme for the frequency domain which does not require physical embedment of watermark in the host image. In other words, the host image will not be affected by the copyright protection scheme and thus the scheme is suitable for protecting precious image work where the host image should be kept original. Multi-copies of unruly black and white binary images would be created by manipulating the characteristics of the discrete cosine transform (DCT) coefficients of host image to XOR with the binary watermark. These unruly copies would then be time-stamped with a digital signature and registered with a certified authority for protecting the rightful ownership. Various attacks like blurring, rescale, cropping, sharpening, noise, rotation, and JPEG lossy compression will be conducted to prove the robustness of the copyright protection scheme. The registered multi-copies of unruly black and white binary images will be used with the attacked host image to generate multi-copies of candidate watermarks. A giveup-table which contains possible attacked host image locations will be employed to help choose the better candidate watermark bits to recover the watermark from the attacked image.

This paper is organized as follows. Related research will be reviewed in Section 2. The proposed copyright protection scheme in the DCT domain will be discussed in Section 3. The experimental results will be analyzed in Section 4.

Finally, discussions and conclusions will be presented in Section 5.

2. Related Research

Traditional image copyright protection scheme usually involves watermark embedment in the host image. The watermark embedding technique can be either in the spatial or frequency domain. Shih et al. [12] proposed to embed watermark using both spatial and frequency domain techniques. The watermark was split into two parts based on the importance of the data. The important part of watermark was then embedded into the DCT coefficients of the host image (the frequency domain embedding), while the non-important part of watermark was embedded directly into the host image's pixel using the simple LSB substitution (the spatial domain embedding). Together with the two parts of watermark embedding, the hiding payload can be increased. The embedding in the frequency domain makes the important part of watermark robust to common image processing attacks. The non-important part of watermark, however, is fragile to image processing attacks.

Wu et al. [13] proposed to hide a watermark using a vector quantization (VQ) codebook. The codebook was first divided into divisions where each division contains two most similar codewords. A block of host image was encoded to be the index of a certain division. The encoded index was then changed to one of its codeword indices in the same division to indicate an embedment of a watermark bit 0 or 1. The final host image with embedded watermark was in its VQ compressed format and thus would save communication bandwidth when transmitted on the Internet. The VQ compression however, will downgrade the host image's quality.

On the other hand, Lin et al.'s scheme [9] hides the binary watermark in the DCT domain. First, the host image was divided into non-overlapping blocks of size 8×8 pixels and then DCT transformed individually. Each block's frequency coefficients consist of {*DC*, AC_1, AC_2, …, AC_{63}}. Here *DC* is the lowest frequency coefficient and is the most significant characteristic of the image. Next, the binary watermark was divided into non-overlapping block of size 2×2. The four bits in each watermark block were then embedded in the least significant bit of AC_3, AC_4, AC_5 and AC_6 coefficients of each 8×8 DCT block. This scheme is not robust against blurring, sharpening and JPEG lossy compression.

3. The Proposed Method

In this paper, an unorthodox image copyright protection scheme is proposed for the DCT frequency domain which does not require physical replacement of the coefficients for hiding the watermark data. In other words, the host image will not be altered by the watermarking scheme as in traditional hiding schemes.

In the proposed scheme, an original *N*×*N* image is first partitioned into *M*×*M* blocks, where each block contains *S*×*S* pixels. Here, the size of *M*×*M* must be the same as the size of the binary watermark. The *S*×*S* sized blocks will be scrambled in a random order, followed by discrete cosine transformation of each individual block (e.g., *S* = 8 as in Fig.1). The *DC* coefficients from all the blocks will be extracted to create an *M*×*M*-sized array (*DC*-array). The *DC*-array will be manipulated to create an *M*×*M*-sized black and white image (referred as *BW*) which will be XORed with the binary watermark (also *M*×*M*-sized) to create multiple copies of unruly black and white differenced images (referred as BW^{dif} as in Fig. 2). The multiple copies are created by XOR with the watermark at different locations of *BW*.

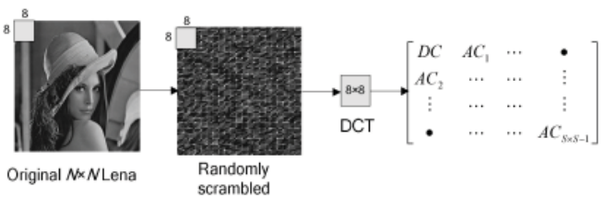


Figure 1. Discrete cosine transformation (DCT)

The result will be certified by the certification authority with a time-stamped digital signature which will be kept in safekeeping by the legalized user. The flow is illustrated as in Fig. 2.

3.1 Encoding process

The image used in this paper is an *N*×*N* sized 8-bits gray-level image *H*. Suppose the black-white binary watermark is *W* of size *M*×*M*. The original image and watermark are defined as in the following equations.

$$\begin{aligned} &H = \{h(i,j) \mid 0 \le i < N,\ \ 0 \le j < N\}, \\ &h(i,j) \in \{0,1,...,255\}, \\ &W = \{w(i,j) \mid\ \ 0 \le i < M,\ \ 0 \le j < M\}, \\ &w(i,j) \in \{0,1\}. \end{aligned} \tag{1}$$

The encoding process is as follows:

Step 1:

Partition the original image *H* into non-overlapping blocks where each block contains *S*×*S* pixels. Altogether there should be *M* × *M* blocks (i.e., $\frac{N \times N}{S \times S} = M \times M$). Next, a seed *K* is used in the random number generator. Image *H*′ is then generated by randomly sequencing the M × M blocks. Equation (2) defined the randomly sequenced image *H*′ consisting of blocks B'(i, j)'s.

$$H' = \{B'(i,j) \mid 0 \le i < M,\ \ 0 \le j < M\}. \tag{2}$$

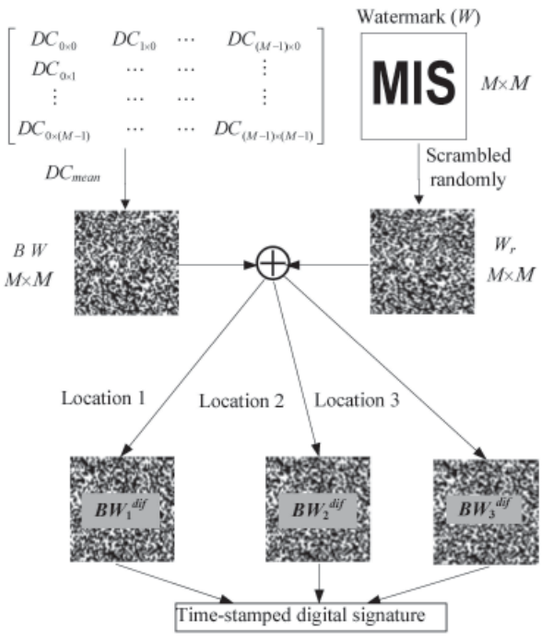


Figure 2. Encrypting the watermark

Step 2:

Perform discrete cosine transformation on each block $B'(i,j)$. Each $B'(i,j)$ will then be transformed to $\hat{B}'(i,j)$ which contains the coefficients DC, AC_1, AC_2, …, $AC_{(s\times s)-1}$. The transformed result is $\hat{H}'$ as shown in equation (3).

$$\hat{H}' = \{\hat{B}'(i,j) \mid 0 \le i < M, 0 \le j < M\},$$
$$\hat{B}'(i,j) = \{DC, AC_1, AC_2, ..., AC_{(s\times s)-1}\}. \quad (3)$$

Step 3:

Extract the DC coefficient from each $\hat{B}'(i,j)$ block and combine all the DC's to make an $M\times M$ sized DC-array H''. The DC-array is shown in equation (4).

$$H'' = \begin{bmatrix} DC_{0\times 0} & DC_{1\times 0} & \cdots & DC_{(M-1)\times 0} \\ DC_{0\times 1} & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \vdots \\ DC_{0\times (M-1)} & \cdots & \cdots & DC_{(M-1)\times (M-1)} \end{bmatrix},$$
$$H'' = \{h''(i,j) \mid 0 \le i < M, 0 \le j < M\}. \quad (4)$$

Step 4:

Calculate the DC_{mean} from DC-array H'' with equation (5). Next, all the DC's in H'' are compared with the DC_{mean} to generate an $M\times M$ sized black and white image BW as shown in equation (6).

$$DC_{Mean} = \frac{\sum_{i=0}^{M-1}\sum_{j=0}^{M-1}(DC_{i\times j})}{M\times M}. \quad (5)$$

$$bw(i,j) = \begin{cases} 1, & \text{if } h''(i,j) > DC_{Mean} \\ 0, & \text{if } h''(i,j) \le DC_{Mean} \end{cases},$$
$$BW = \{bw(i,j) \mid 0 \le i < M, 0 \le j < M\}, \quad (6)$$
$$bw(i,j) \in \{0,1\}.$$

Step 5:

Randomly scramble the binary watermark W (using seed K) to generate a scrambled watermark W_r such that
$W_r = \{ w_r(i,j) \mid 0 \le i < M, 0 \le j < M \}$, $w_r(i,j) \in \{0, 1\}$.

Step 6:

Perform XOR on W_r with BW at different locations to generate multiple unruly black and white differenced images. In our paper, we only generated three differenced binary images { BW_1^{dif}, BW_2^{dif}, BW_3^{dif} }. The positions of BW to XORed with W_r are defined in equations (7), (8) and (9), respectively.

$$BW_1^{dif} = \{bw_1^{dif}(i,j) \mid 0 \le i < M, 0 \le j < M\},$$
$$bw_1^{dif}(i,j) \in \{0,1\}, \quad (7)$$
$$bw_1^{dif}(i,j) = w_r(i,j) \oplus bw(i_1, j_1),$$

where $i_1 = i, j_1 = j$.

$$BW_2^{dif} = \{bw_2^{dif}(i,j) \mid 0 \le i < M, 0 \le j < M\},$$
$$bw_2^{dif}(i,j) \in \{0,1\}, \quad (8)$$
$$bw_2^{dif}(i,j) = w_r(i,j) \oplus bw(i_2, j_2),$$

where

$i_2 = (i + M/2) \bmod M$, $j_2 = (j + M/2) \bmod M$.

$$BW_3^{dif} = \{bw_3^{dif}(i,j) \mid 0 \le i < M, 0 \le j < M\},$$
$$bw_3^{dif}(i,j) \in \{0,1\}, \quad (9)$$
$$bw_3^{dif}(i,j) = w_r(i,j) \oplus bw(i_3, j_3),$$

where

$i_3 = (i + M - 1) \bmod M$, $j_3 = (j + M - 1) \bmod M$.

The authorized users must have in possession the time-stamped and certified three copies of { BW_1^{dif}, BW_2^{dif}, BW_3^{dif} }, random seed K and DC_{mean} for recovering the watermark. Let N_0 be the number of bits valued 0 in the original watermark. Let N_1 be the number of bits valued 1 in the original watermark. Let $N_x \ge N_y$, where $x,y \in \{0,1\}$ and $x \ne y$. A simple calculation is used to set a bit-flag mb equals x. In other words, mb is the bit value of the most bit value (either "0" or "1") in the original watermark. This bit-flag mb will also be sent to the authorized user for later usage in recovering the watermark.

3.2 Restoring the watermark

Decoding the watermark from attacked host image will require repeating steps in the encoding procedure. An $M\times M$ sized black and white image BW_A is generated by replicating Step 1 to Step 4 in the encoding process using the DC_{mean} previously kept by the authorized user. BW_A is then XORed with {BW_1^{dif}, BW_2^{dif}, BW_3^{dif}} to obtain three copies of candidate watermark {W_1^A, W_2^A, W_3^A}. A giveup-table which contains possible attacked locations will be employed to help recover the watermark from the three copies of {W_1^A, W_2^A, W_3^A}.

3.2.1 Giveup-table

A giveup-table was employed as containment for restoring watermark under attacks. A typical mono-gray 8×8 DCT block contains a single value in DC and zeroes in all the AC's. The image blocks in the cropped area have the same phenomenon as the mono-gray blocks. Therefore, an $M\times M$ bitmapped giveup-table H_A^{giveup} is used to record the monochromatic pattern of an image. "0" is used to represent the non-mono-gray block and "1" otherwise. Thus H_A^{giveup} is defined as in equation (10).

$$h_A^{giveup}(i,j) = \begin{cases} 1, & \text{if } \hat{B}_A'(i,j) \text{ contains only one } DC \text{ value} \\ 0, & \text{otherwise} \end{cases},$$
$$H_A^{giveup} = \{h_A^{giveup}(i,j) \mid 0 \le i < M, 0 \le j < M\}, \quad (10)$$
$$h_A^{giveup}(i,j) \in \{0,1\}.$$

In equation (10), $\hat{B}_A'(i,j)$ is the same as $\hat{B}'(i,j)$ in equation (3) except that it is generated from the attacked image H_A.

When recovering the watermark, if a bit value $h_A^{giveup}(i,j)$ equals to 1, this means that its corresponding image block at location (i,j) could be attacked by cropping or it is a mono-gray block. The giveup-table is then used to reference the locations of possible attacked blocks. In the proposed scheme, each of the 3 differenced binary images {BW_1^{dif}, BW_2^{dif}, BW_3^{dif}} should match the same locations in the watermark. Therefore, the giveup-table H_A^{giveup} can be used to help remove the unwanted locations. A more accurate watermark could then be recovered based on locations that do not belong to the giveup locations.

3.2.2 Process in restoring the watermark

Suppose there is an attacked image H_A. The following are steps used in recovering the watermark W_A from attack.

Step 1:

Repeat Step 1 to Step 4 in the encoding process. The *DC*'s calculated from the H_A blocks are recorded in the *DC*-array H_A". The only difference here from the encoding process is that the *DC* values are compared with the DC_{mean} previously sent to the authorized user rather than with the mean calculated from the attacked image. We will get a black and white image BW_A.

Step 2:

Perform XOR operation on BW_A with the three copies of $\{BW_1^{dif}, BW_2^{dif}, BW_3^{dif}\}$ using equations (11), (12) and (13), respectively. This will generate three candidate watermarks $\{W_1^A, W_2^A, W_3^A\}$.

$$W_1^A = \{w_1^A(i,j) \mid 0 \le i < M, 0 \le j < M\},$$
$$w_1^A(i,j) \in \{0,1\},$$
$$w_1^A(i,j) = bw_1^{dif}(i,j) \oplus bw_A(i_1, j_1), \quad (11)$$

where $i_1 = i, j_1 = j$.

$$W_2^A = \{w_2^A(i,j) \mid 0 \le i < M, 0 \le j < M\},$$
$$w_2^A(i,j) \in \{0,1\},$$
$$w_2^A(i,j) = bw_2^{dif}(i,j) \oplus bw_A(i_2, j_2), \quad (12)$$

where

$$i_2 = (i + M/2) \bmod M,\ j_2 = (j + M/2) \bmod M.$$

$$W_3^A = \{w_3^A(i,j) \mid 0 \le i < M, 0 \le j < M\},$$
$$w_3^A(i,j) \in \{0,1\},$$
$$w_3^A(i,j) = bw_3^{dif}(i,j) \oplus bw_A(i_3, j_3), \quad (13)$$

where

$$i_3 = (i + M - 1) \bmod M,\ j_3 = (j + M - 1) \bmod M.$$

Step 3:

Generate the giveup-table H_A^{giveup} using equation (10).

Step 4:

Perform a mapping of $\{W_1^A, W_2^A, W_3^A\}$ with H_A^{giveup} using a simple table look-up operation to choose qualified candidate watermark bits in $\{W_1^A, W_2^A, W_3^A\}$. If a bit "0" is found in a location of H_A^{giveup} then its corresponding watermark bit in $\{W_1^A, W_2^A, W_3^A\}$ will be classified as qualified candidate. On the other hand, if a bit "1" is found, its corresponding watermark bit in $\{W_1^A, W_2^A, W_3^A\}$ will be discarded. Mapping for $\{W_1^A, W_2^A, W_3^A\}$ and the H_A^{giveup} is done in equations (14), (15) and (16).

$$\text{if} \begin{cases} h_A^{giveup}(i_1, j_1) = 0 \\ h_A^{giveup}(i_1, j_1) = 1 \end{cases} \text{then} \begin{matrix} \text{keep } w_1^A(i,j) \\ \text{giveup } w_1^A(i,j) \end{matrix}, \quad (14)$$

where $i_1 = i, j_1 = j$.

$$\text{if} \begin{cases} h_A^{giveup}(i_2, j_2) = 0 \\ h_A^{giveup}(i_2, j_2) = 1 \end{cases} \text{then} \begin{matrix} \text{keep } w_2^A(i,j) \\ \text{giveup } w_2^A(i,j) \end{matrix}, \quad (15)$$

where

$$i_2 = (i + M/2) \bmod M,\ j_2 = (j + M/2) \bmod M.$$

$$\text{if} \begin{cases} h_A^{giveup}(i_3, j_3) = 0 \\ h_A^{giveup}(i_3, j_3) = 1 \end{cases} \text{then} \begin{matrix} \text{keep } w_3^A(i,j) \\ \text{giveup } w_3^A(i,j) \end{matrix}, \quad (16)$$

where

$$i_3 = (i + M - 1) \bmod M,\ j_3 = (j + M - 1) \bmod M.$$

Step 5:

Recover the watermark bits by the following case analysis. There could be 0~3 candidates for each watermark bit at location (*i,j*). The different cases are analyzed as follows.

Case 1: candidate = 3 (all three are qualified candidates).

Polling method is used to select the bit in $\{w_1^A(i,j), w_2^A(i,j), w_3^A(i,j)\}$ as the recovered watermark bit. Suppose $w_1^A(i,j)$=1, $w_2^A(i,j)$=0 and $w_3^A(i,j)$=1, then "1" is the recovered bit.

Case 2: candidate = 2 (only two are qualified candidates).

From $\{w_1^A(i,j), w_2^A(i,j), w_3^A(i,j)\}$, only two candidates qualify. For the two qualified candidate, the differenced distance is calculated for DC_{mean} and *DC*-array H_A" at the same location (*i,j*). The further the distance the less possible it is to be affected by an attack because it is more difficult to change a greater distance to reverse the corresponding $bw_A(i,j)$ bit (see equation (6)). Suppose $w_1^A(i,j)$=1 and $w_2^A(i,j)$=0 are the two candidates that qualify. D_1 and D_2 are calculated as in equations (17) and (18). If $D_1 > D_2$ then $w_1^A(i,j)$=1 is the recovered watermark bit. Equations (17), (18) and (19) are used to calculate the differenced distance.

$$D_1 = |h_A"(i,j) - DC_{mean}|, \quad (17)$$

distance corresponding $w_1^A(i,j)$.

$$D_2 = |h_A"((i + M/2) \bmod M, (j + M/2) \bmod M) - DC_{mean}|, \quad (18)$$

distance corresponding $w_2^A(i,j)$.

$$D_3 = |h_A"((i + M - 1) \bmod M, (j + M - 1) \bmod M) - DC_{mean}|, \quad (19)$$

distance corresponding $w_3^A(i,j)$.

Case 3: candidate = 1 (only one qualified candidate).

The only one qualified candidate in $\{w_1^A(i,j), w_2^A(i,j), w_3^A(i,j)\}$ is the bit used to recover the corresponding watermark bit at location (*i,j*).

Case 4: candidate = 0 (no qualified candidate).

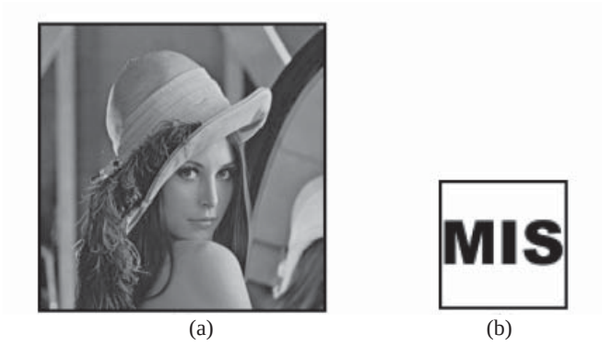
When there is no qualified candidate, the recovered watermark bit is set to the bit-flag *mb* previously sent to the authorized user. The bit-flag *mb* is the value of the original watermark bit that represents the most bit value (either "0" or "1") in the original watermark.

The recovered watermark from the above case analyses is a scrambled watermark and will require the seed value *K* to restore to its unscrambled state W_A.

4. Experimental results

For the experimental tests, a 512×512 gray image Lena and, a 64×64 black and white watermark will be used (see Fig.3). Normalized correlation (*NC*), as calculated in equation (20) where $w_A(i,j)$ represents the watermark bit recovered from attacks, is used to estimate the similarity of the recovered watermark to its original. In equation (20), bit "1" is used to represent black colored pixel in the black-white watermark. Thus, the larger the *NC* is, the clearer the recovered watermark will be.

$$NC = \frac{\sum_{i=0}^{M-1}\sum_{j=0}^{M-1} w(i,j)\, w_A(i,j)}{\sum_{i=0}^{M-1}\sum_{j=0}^{M-1} \left[w(i,j) \right]^2}, \quad NC \in [0,1]. \quad (20)$$



(a) (b)

Figure 3. (a) Original Lena (512×512), (b) Black and white watermark (64×64)

Figure 4. Restoring watermark after Gaussian blurring attack

Testing will be conducted on different attacks such as blurring, rescale, cropping, sharpening, noise, rotation, and JPEG lossy compression using the image software tool Photoimpact 11 to prove the robustness of the recovered watermark.

4.1 Gaussian blurring attacks

The Gaussian blurring attack is conducted for a radius from 1 to 5 pixels as in the tool Photoimpact 11. Gaussian blurring is a low-pass filter that reduces the edge sharpness of an image. As seen in Table 1, *NC* is 0.998 for a radius of 1 and 0.992 for a radius of 5. Fig.4(b) shows Lena as perceptually blur from the attacks; but the recovered watermark in Fig.4(c) is clear with *NC*=0.992. This proves that the proposed copyright protection scheme is robust against the blurring attack.

Blur (pixel)	**1**	**2**	**3**	**4**	**5**
NC	0.998	0.997	0.996	0.994	0.992

Table 1. Results from Gaussian blurring attacks on Lena

4.2 Rescale attacks

In the rescale attack, the original image is rescaled from 512×512 pixels to 256×256 pixels and vice versa. As seen in Fig. 5, the rescaled NC for the watermark is 0.999 and the restored watermark is perceptibly clear; thereby, proving that our scheme is robust to rescale attack.

4.3 Cropping attacks

As shown in Table 2, the restored watermark can still achieve *NC*>0.86 after cropping 50% of Lena. In Fig. 6(c), the restored watermark is still visually perceptible after cropping 80%.

Figure 5. Rescale from 512×512 to 256×256 and vice versa

The proposed copyright protection scheme showed strong robustness against the cropping attack largely due to the three copies of the differenced binary BW^{dif}s (i.e., BW_1^{dif}, BW_2^{dif}, and BW_3^{dif}) which were purposely XORed at different locations.

Crop	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**
NC	0.993	0.963	0.938	0.915	0.866	0.792	0.691	0.498	0.296

Table 2. Results from cropping attacks on Lena

Figure 6. Cropping attacks (10%, 50%, 80%, and 90%)

In comparison to Wu et al.'s method [13] which achieves a *NC*=0.749 for cropping at 25% and *NC*=0.497 for cropping at 50%, our method can achieve *NC*=0.952 for cropping at 25% and *NC*=0.866 for cropping at 50%. As seen in Fig. 7, the restored watermarks are perceptibly clear in our methods whereas in Wu et al.'s method the recovered watermarks were covered with background noises. The noises were denser when cropping is 50%. Their experimental testing stopped at 50% cropping. However, our method showed that at 80% cropping, still, the watermark can be recovered and the background did not suffer from dense background noises as seen in 50% cropping for Wu et al.

4.4 Sharpening attacks

Sharpening is a high-pass filter used to filter out the smooth area of an image to make the image contour distinct. Fig. 8 shows the recovered watermark with the original image sharpened at levels 20, 40, 60, 80 and 100, respectively, as in the tool

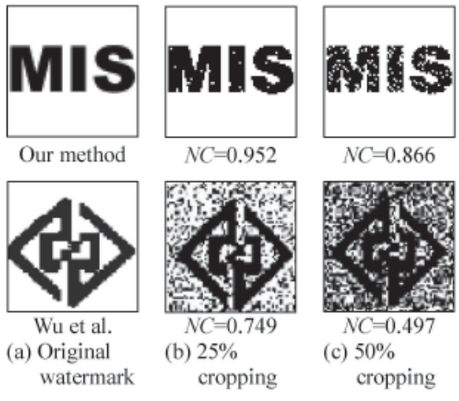


Figure 7. Comparing cropping attacks with Wu et al. [13] and our method

Figure 8. Sharpening attacks at levels (a) 20, (b) 40, (c) 60, (d) 80, and (e) 100

Photoimpact 11. At level 100, *NC*=0.975 for the retrieved watermark. As shown in Fig. 8(e), the watermark is perceptibly clear and has only minor background noises.

4.5 Uniform noise attacks

Table 3 illustrates results from uniform noise attacks with noise variances from 10 to 100 as in the tool Photoimpact 11. From Table 3, *NC*=0.999 for noise of 10 and *NC*=0.957 for noise of 100. As seen in Fig. 9(b), the recovered watermark for Lin et al.'s method [9] is distorted with *NC*=0.806 from attack noise of 10. On the other hand, our method for the same noise in Fig. 9(c) showed clear recovered watermark with *NC*=0.999. Fig. 9(d) shows our method has a clear recovered watermark with *NC*=0.957 at attack noise of 100 – it performed better than Lin et al.'s at noise of 10.

Noise	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
NC	0.999	0.998	0.998	0.996	0.995	0.991	0.984	0.975	0.965	0.957

Table 3. Results from uniform noise attacks on Lena

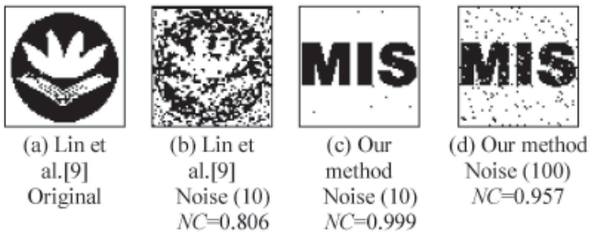


Figure 9. Comparing uniform noise attacks with Lin et al. [9] and our method

4.6 Rotation attacks

Rotation and resizing is a geometrical rotational transformation of an image. The center is regarded as the basic point where an image is tilted at an angle in the rotational transformation. The rotation attack is conducted for rotations in five different degrees (1°, 2°, 3°, 4° and 5°) in a clockwise direction and each time resized at the rotated angle. As shown in Fig. 10, at 5° the watermark has an *NC*=0.844 and is perceptibly clear with minor background noises. The rotation attack is unaddressed in most researches in watermark techniques. Most techniques are fragile to rotation. Nonetheless, in our method the watermark is robust at 5° rotation.

Figure 10. Rotation attacks at (a) 1°, (b) 2°, (c) 3°, (d) 4°, and (e) 5°

4.7 JPEG lossy compression attacks

As seen in Table 4, comparisons between Wu et al.'s [13] and our methods for JPEG lossy compression at compression ratios 10.2, 12.0 and 14.1 showed that our method performed better than Wu et al.'s with *NC*s at 0.999, 0.998, 0.998 for our method and, 0.949, 0.913 and 0.876 for Wu et al.'s, respectively. Wu et al.'s recovered watermarks are distorted and covered with background noises while our watermarks are clear and have little noises; the same is true for our watermark at ratio 21.0 (not shown in Wu et al.'s). Table 5 showed comparisons between our method and Lin et al.'s [9] method. Lin et al.'s recovered watermark is extremely distorted at JPEG ratio 10 with *NC*=0.736. In comparison, our proposed method is more robust.

Ratios	(a) 10.2	(b) 12.0	(c) 14.1	(d) 21.0
Our method	NC=0.999	NC=0.998	NC=0.998	NC=0.973
Wu et al.	NC=0.949	NC=0.913	NC=0.876	none

Table 4. Comparing our method vs. Wu et al. [13] for JPEG lossy compression

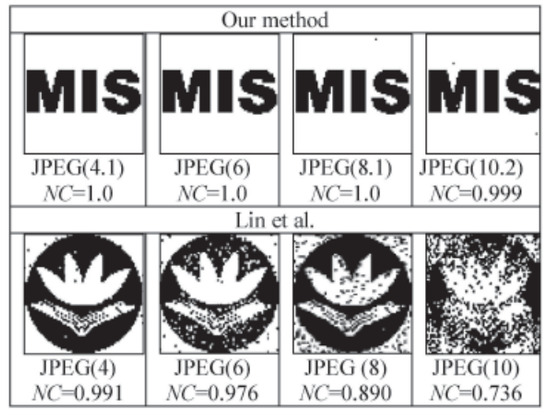


Table 5. Comparing our method vs. Lin et al. [9] for JPEG lossy compression

Conclusions

In most watermark hiding techniques, the watermark is hidden in the images either in the spatial or frequency domain. The host images will suffer a certain degree of distortions for the embedment. The proposed image copyright protection scheme does not physically alter the host image. Although it will take about 1.5 K bytes of storage (64×64×3 bits) to keep the three copies of differenced binary BW^{dif}s, the host image never suffers any data loss from the watermark process. Thus, the proposed copyright protection scheme is suitable for protecting precious image work where the host image should not be altered.

Results from the experimental testing on the seven different attacks showed that the recovered watermarks are visually clear and have little background noises. All have *NC*>0.95 with the exceptions that *NC*>0.86 for up to 50% cropping and *NC*>0.84 for 5° rotation attack. These results proved that the proposed copyright protection technique is very robust.

References

[1] Briassouli, A., Tsakalides, P., Stouraitis, A. (2005). Hidden Messages in Heavy-tails: DCT-domain Watermark Detection Using Alpha-stable Models. *IEEE Transactions on Multimedia*, 7 (4) 700-715.

[2] Celik, M.U., Sharma, G., Tekalp, A.M. (2006). Lossless Watermarking for Image Authentication: A New Framework and an Implementation. *IEEE Transactions on Image Processing*, 15 (4) 1042-1049.

[3] Chang, C.C., Hu, Y.S., Lu, T.C. (2006). A Watermarking-based Image Ownership and Tampering Authentication Scheme. *Pattern Recognition Letters*, 27 (5) 439-446.

[4] Chang, C.C., Tsai, P.Y., Lin, M.H. (2005). SVD-based Digital Image Watermarking Scheme. *Pattern Recognition Letters*, 26 (10) 1577-1586.

[5] Chen, J., Chen, T.S., Chen, J.G. (2004). Efficient Sub-band Coding in JPEG2000 for Improved Authentication and Tamper-proofing. *Imaging Science Journal*, 52 (3) 181-187.

[6] Chen, J., Chen, T.S., Hsu, H.C., Chen, H.W. (2005). New Visual Cryptography System Based on Circular Shadow Images and Fixed Angle Segmentation. *Journal of Electronic Imaging*, 14 (3) 033018:1-15.

[7] Chu, W.C. (2003). DCT-based Image Watermarking Using Subsampling. *IEEE Transactions on Multimedia*, 5 (1) 34-38.

[8] Lu, Z.M., Xu, D.G., Sun, S.H. (2005). Multipurpose Image Watermarking Algorithm Based on Multistage Vector Quantization. *IEEE Transactions on Image Processing*, 14 (6) 822-831.

[9] Lin, S.F.D., Chen, C.F. (2000). A Robust DCT-based Watermarking for Copyright Protection. *IEEE Transactions on Consumer Electronics*, 46 (3) 415-421.

[10] Potdar, V.M., Han, S., Chang, E. (2005). A Survey of Digital Image Watermarking Techniques. *In: Proc. of the 3rd IEEE International Conference on Industrial Informatics (INDIN'05)*, pages 709-716, Aug. 10-12, 2005.

[11] Raval, M.S., Rege, P.P. (2003). Discrete Wavelet Transform Based Multiple Watermarking Scheme. *In: Proc. of the 2003 Conference on Convergent Technologies for Asia-Pacific Region (TENCON'03)*, 3, pages 935-938, Oct. 15-17, 2003.

[12] Shih, F.Y., Wu, S.Y.T. (2003). Combinational Image Watermarking in the Spatial and Frequency Domain. *Pattern Recognition*, 36, 969-975.

[13] Wu, H.C., Chang, C.C. (2005). A Novel Digital Image Watermarking Scheme Based on the Vector Quantization Technique. *Computers & Security*, 24 (6) 460-471.

[14] Yuan, Y., Huang, D., Liu, D. (2006). An Integer Wavelet Based Multiple Logo-watermarking Scheme. *In: Proc. of the First International Multi-Symposiums on Computer and Computational Sciences (IMSCCS'06)*, 2, pages 175-179, April 20-24, 2006.

Authors biography

Chi-Nan Lin received the B.S. degree in Psychology from the National Chengchi University, Taiwan, in 1982. He received the M.S. degree in Computer Science from the University of Louisiana, USA, in 1985. He is currently working towards the Ph.D. degree in Computer Science and Information Engineering at National Chung Cheng University, Taiwan. His field of interest is in information hiding and image security. Currently he is also a faculty member of the Department of Management Information Systems at Central Taiwan University of Science and Technology, Taiwan.

Chin-Chen Chang received his B.S. degree in applied mathematics in 1977 and the M.S. degree in computer and decision sciences in 1979, both from the National Tsing Hua University, Hsinchu, Taiwan. He received his Ph.D. in computer engineering in 1982 from the National Chiao Tung University, Hsinchu, Taiwan. Since February 2005, he has been a Chair Professor of Feng Chia University. In addition, he has served as a consultant to several research institutes and government departments. His current research interests include database design, computer cryptography, image compression and data structures.

Mien-Tsung Tsai received the B.S. degree in Mechanical Engineering from the National Chung-Hsing University, Taiwan, in 1991. He received the M.S. degree in Mechanical Engineering from The National Taiwan University of Science and Technology, Taiwan, in 1993. He is currently working towards the Ph.D. degree in Computer Science and Information Engineering at National Chi-Nan University, Taiwan. Currently His research interests are in the digital library and aesthetic computing.