Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105 (Online)
Volume **3**, Number **4**, December **2018**

A DCT-based Robust Watermarking Scheme Surviving JPEG Compression with Voting Strategy

Pei-Feng Shiu[1], Chia-Chen Lin[2,*], Jinn-Ke Jan[2], Ya-Fen Chang[3]

[1]Department of Computer Science and Engineering,
National Chung Hsing University
250 Kuo Kuang Rd., Taichung 402, Taiwan
d100056005, jkjan@cs.nchu.edu.tw

[2]Department of Computer Science and Information Management,
Providence University, 200 Chung-Chi Rd., Taichung, Taiwan
*Corresponding author: mhlin3@pu.edu.tw

[3]Department of Computer Science and Information Engineering,
National Taichung University of Science and Technology,
No. 129, Sec. 3, Sanmin Rd., Taichung, Taiwan
cyf@nutc.edu.tw

Received September 2017; revised June 2018

Abstract. *In this paper, a robust watermarking scheme based on DCT for image copyright protection is proposed. The proposed scheme uses the concept of mathematical remainder to modify the DCT coefficient to ensure its robustness against incidental attacks and malicious attacks. A voting strategy is applied in this scheme to enhance the robustness of the watermark. Experimental results confirm that the robustness of a hidden watermark against JPEG compression with our proposed scheme is better than that of Lin et al.'s and Patra et al.'s schemes.*
Keywords: Robust watermarking, DCT, JPEG, Proof of ownership.

1. **Introduction.** Digital watermarking is a process used to hide data into cover media, such as images or video streams. It can be used to convey information secretly or to protect the copyright and integrity of the cover medium itself [1, 22]. Digital image watermarking can be either visible or invisible. Visible watermarking contains visible information, such as a company logo, to indicate the owner of that multimedia. However, visible watermarking causes large distortion of the cover image, and hence invisible watermarking is more practical. In the case of invisible digital watermarking, no visual artifact is expected in the watermarked image. In other words, usually this watermark is imperceptible in a watermarked image. Invisible watermarking can be classified into three types: robust, fragile and semi-fragile watermarking's. Robust watermarking [2-4] is mainly applied to broadcast monitoring, proof of ownership, transaction tracking, and copy control since the watermark is robust for malicious tampering. Fragile [5-7] or semi-fragile [8-10] watermarking is generally used for content authentication as a result of this watermark being fragile to any slight modification of the watermarked image. Robust watermarking is primarily applicable to broadcast monitoring, proof of ownership, transaction tracking, and copy control to prevent the unauthorized copying of digital media. Robustness describes how well the watermark survives common signal processing operations. For example, in applications where we have to detect a watermark in a copy

of a medium that has been transferred over an analog channel, that watermark must be robust against the noise channel. There are three essential requirements for robust watermarking. The first is robustness, that is, the watermark should be robust enough to withstand all kinds for signal processing operations, attacks, or unauthorized access. Any attempt with the potential to alter the data content is considered as an attack. The success of robust watermarking for copyright protection depends on its robustness against attack. The second requirement of a watermark is fidelity, namely, a perceptual similarity between the original and the watermarked versions of the media. Fidelity is the most fundamental requirement for any watermarking scheme. The watermark will only show up on the watermark detector device. The third requirement for success is security, that is, the watermark must only be accessible to authorized parties and only authorized parties are allowed to alter the watermarked media. Encryption can also be used to prevent unauthorized access of the watermarked media in general terms. Most robust watermarking schemes treat certain practical manipulations, such as image compression and image processing, as attacks. For digitized images to be safely and efficiently transmitted over the Internet, therefore, watermarked images should be particularly robust to JPEG compression. To improve the robustness of a watermark, most of the recently proposed techniques embed the watermarks into the low frequency part of images. Huang et al. [11] proposed a watermarking scheme based on a quantitative analysis of the magnitudes of DCT components of host images. The authors claim that more robustness can be achieved if watermarks are embedded in DC components since DC components have a much larger perceptual capacity than any AC component does. In addition, the feature of texture masking and luminance masking of the human visual system are incorporated into watermarking. Lin et al. [12, 13] proposed two kinds of DCT-based image watermarking techniques, both of which perform well under general JPEG compression. Generally speaking, embedding watermark into low-frequency domain could make watermark more robust. However, the larger variation of low-frequency coefficient is, the worse the image quality will be. In Lin et al.'s [12] scheme, they embedded watermark bits by replacing the least-significant bit (LSB) of DCT coefficients in low-frequency. It reduces the variation of DCT coefficients and remains the image quality of watermarked image and the robustness of hidden watermark. Later, Lin et al. [13] proposed another scheme to further improve the robustness of the hidden watermark. In Lin et al.'s scheme [13], they tried to reduce the influence of hidden watermark by using a pre-determined threshold when the DCT coefficients are modified for watermark embedding. However, when these watermarked images have to be compressed to a higher compression ratio, the embedded watermarks may be destroyed seriously. To overcome this problem, Lin et al. [14] proposed a DCT-based image watermarking technique to improve the robustness of watermarks against JPEG compression. Directly replacing low- frequency components with a watermark may introduce undesirable degradation to image quality. Thus, their scheme adjusts the DCT low frequency coefficients using the concept of mathematical remainder to preserve acceptable visual quality for watermarked images. Under the same circumstance, Lin et al.'s scheme [14] has better robustness against JPEG compression attack than their early works [12, 13]. The Chinese remainder theorem (CRT) [15] has been used in several engineering applications, including RSA algorithm, secret sharing, polynomial interpolation theory, residue number systems, and prime-factor fast Fourier transform. Based on its CRT properties, two CRT-based watermarking schemes with preliminary results are reported in [16, 17]. The two schemes embed watermarks based on the Discrete Cosine Transform (DCT) technique. The use of CRT provides additional security along with further resistance to certain familiar attacks. Since CRT involves

only modular operations for its computation, the time required for embedding and extraction of a watermark for the proposed schemes are much less when compared to the SVD-based scheme. However, the two schemes cannot withstand image manipulations and JPEG compression quite as well. Our proposed scheme attempts to overcome the problems faced in [14, 16-17]. To maintain the visual quality of watermarked images, only low-frequency DCT coefficients are selected to carry hidden watermarks using the concept of mathematical remainder. A voting strategy is applied to enhance the robustness. To test the robustness, we considered several attacks, including cropping, tampering, noise, brightening, sharpening, and JPEG compression. The proposed scheme does achieve a high peak signal to noise ratio (PSNR) and also low tamper assessment function (TAF) values for most of the attacks. To have this paper self-contained, Section II introduces the concept of Discrete Cosine Transform, Chinese Reminder Theorem, Patra et al. scheme 2 [17] and Lin et al. scheme [14] which will be used to compare with our proposed scheme in Section IV. Section III contains a detailed exposition of the proposed algorithm. In Section IV, we experimentally investigate the relationship between the capacity and distortion, and the influence of variant attacks on robustness. We also compare performance to existing watermarking schemes in the same section. Finally, we conclude the paper in Section V.

2. Related Works.

In this section, Chinese Remainder Theorem which used to design our proposed watermarking scheme; Lin et al. [14] and Patra et al. [17] schemes which used to mainly compare with the performance of our proposed watermarking scheme in PSNR and TAF under various attacks will be introduced, respectively, in the following two subsections.

2.1. Discrete Cosine Transform.

The Discrete Cosine Transform (DCT) is a widely application used for image transformation adapted to compress JPEG images. For 8 8 pixels block 2-dimensional DCT formula is given below:

$$DCT(i,j) = \frac{c(i)c(j)}{4} \sum_{x=0}^{7} \sum_{y=0}^{7} pixel(x,y) \times \cos(\frac{(2x+1)i\pi}{16}) \cos(\frac{(2y+1)i\pi}{16}), \quad (1)$$

where c(i)c(j) = $\frac{1}{\sqrt{2}}$ if i,j=0 otherwise c(i)c(j)=1.

There $DCT(i,j)$ and $pixel(x,y)$ present a DCT coefficient at the position (i,j) and a pixel value at the position (x,y), respectively. When the DCT coefficients of image should be transformed into pixel values, the 2-dimensional inverse DCT formula will be used as follows:

$$pixel(x,y) = \frac{c(i)c(j)}{4} \sum_{x=0}^{7} \sum_{y=0}^{7} DCT(i,j) \times \cos(\frac{(2i+1)i\pi}{16}) \cos(\frac{(2j+1)i\pi}{16}), \quad (2)$$

where c(i)c(j) = $\frac{1}{\sqrt{2}}$ if i,j=0 otherwise c(i)c(j)=1.

2.2. Chinese Remainder Theorem.

The Chinese remainder theorem (CRT) [15] is a result about congruences in number theory and its generalizations in abstract algebra. It states that an integer can be completely described by the sequence of its remainders. Let μ be a set of r integers given by $\mu = M_1, M_2, .., M_r$, such that any two Mi are pairwise relatively prime. The theorem can also be generalized as follows. Given a set of simultaneous congruences as follows:

$$Z \cong R_i (\mathrm{mod} M_i), \quad (3)$$

where $R_i, i = 1, 2, ..., r$ are called residues, and the solution of the set of congruences is defined as follows:

$$Z = \{\sum_{i=1}^{r} R_i \frac{M}{M_i}\} (\mathrm{mod} M_i), \tag{4}$$

where $M = M_1 \times M_2 \times .. \times M_r$, and the K_i are determined from the following formula:

$$K_i \frac{M}{M_i} \cong 1 (\mathrm{mod} M_i), \tag{5}$$

Here, a simple example with $r = 2$ is presented to describe CRT. Let $M_1 = 7, M_2 = 13$. Let the two congruences be given as $Z = 1(mod7)$ and $Z = 9(mod13)$. Thus, $M_1 = 1, M_2 = 9$. In order to find the value of Z, compute $M = M_1, M_2 = 91$. K_1 and K_2 are determined from the following formula:

$$K_1 \frac{91}{7} \cong 1(\mathrm{mod}7) and (K_2 \frac{91}{13}) \cong 1(\mathrm{mod}13), \tag{6}$$

We can see that for $K_1 = -1$ and $K_2 = 2$, these two congruences are satisfied. Now Z is determined as

$$Z \cong \{\sum_{i=1}^{2} R_i \frac{M}{M_i}\} (\mathrm{mod}7 \times 13) = 113 (\mathrm{mod}91) = 22 \tag{7}$$

2.3. **Patra et al. scheme [17].** In 2010, Patra et al. [17] proposed a CRT-based watermark scheme for DCT domain. Their scheme applies CRT to hide watermark at low-frequency area of DCT coefficients. The embedding and extraction procedures are presented in this section

2.3.1. *The CRT application of Patra et al.'s scheme.* From the previous example in CRT, given the value of Z and $r = 2$ for the set μ, the M_1 and M_2 are two integers of set μ. The residues R_1 and R_2 are obtained by using formula (3) of CRT. The absolute difference between R_1 and R_2 can represented as d as follows:

$$d = \lfloor R_1 - R_2 \rfloor \tag{8}$$

And then the find the maximum value of d by taking the lager of the two moduli M_1 and M_2 and decrease one from it. It is represented by D as:

$$D = \max\{M_1, M_2\} - 1 \tag{9}$$

2.3.2. *The watermark embedding procedure.* This process begin form dividing original image into 8×8 pixel blocks. For example, a 512×512 pixel size image will divide to 64× 64 blocks. Then one coefficient of these blocks be using for embedding watermark bits. These steps of embedding watermark are given in below.
Step 1 Divide the image into several non-overlapping 8×8 pixel blocks.
Step 2 Determine the coefficients block according the DCT conversion to the 8×8 pixel blocks.
Step 3 Randomly select a watermark bit hiding into a DCT coefficient block.
Step 4 Select a DCT coefficient randomly either the DC or AC coefficient, and use it to embed the watermark bit. Let the DCT coefficient value be denoted as Z.
Step 5 Let M_1 and M_2 be the pair-wise co-prime numbers use for CRT.
Step 6 Determine R_1 and R_2 apply the formula of CRT (3) with $Z, M_1 and M_2$.
Step 7 Determine d and D using the formulas (8) and (9), respectively.
Step 8 According the watermark bit modify value Z to embedding watermark bit. If watermark bit is '0', the modify condition is:

$$d \geq \frac{D}{c} \tag{10}$$

If watermark bit is '1', the modify condition is:

$$d < \frac{D}{c} \tag{11}$$

where constant $c = 2$ if selected coefficient is DC, otherwise, constant $c = 4$. The parameter d and D are determined at step 7. If the according condition is not satisfied the value Z will be increase (or decrease) 8 until the condition to be satisfied or the change range over the limit 256. Step 9 Determine the modified pixels block by apply the Inverse DCT conversion to the coefficients block. Step 10 Repeat steps 3-9 to modify remaining DCT coefficients blocks until all watermark bits are embedded. Since the range of possible values for the DC coefficients are from 0 to 2040, and the ranges of AC coefficients from -1020 to 1020. According to CRT, the product of the pair-wise co-prime numbers M_1 and M_2 must greater than maximum possible number for the range of coefficients. From the experiments result of Patra et al.'s scheme [17], the most suitable pair-wise co-prime numbers should be 38, 107 and 38, 55, respectively, for DC and AC coefficients. In addition, assuming the change of DCT coefficients too less, the embedding watermark will be "invalid". Thus at step 8, Patra et al. [17] use value 8 to change coefficients and set maximum limit of change as 256.

2.3.3. *The watermark extracting procedure.* Before the extracting process, we need to know some information to extract the embedded watermark such as: watermarked image, watermark size, seed of the PRNG (Pseudo Random Number Generator) and the pair-wise co-prime numbers M_1 and M_2. Follow these steps; the embedded watermark will be reconstructed.
Step 1 Divide the watermarked image into several non-overlapping 8×8 pixel blocks.
Step 2 Determine the coefficients block according the DCT conversion to the 8×8 pixel blocks.
Step 3 Use the seed of PRNG to randomly select coefficients which embedded watermark bit in a block denote as Z'.
Step 4 Use CRT's formula (3) with value M_1, M_2 and $Z'$to determine residues R_1 and R_2.
Step 5 Apply formula (8) with R_1 and R_2 to determine difference value d.
Step 6 Apply formula (9) with M_1 and M_2 to determine D.
Step 7 If formula (10) is satisfied, the watermark bit is '1'; on the other hand, when the condition is not satisfied means watermark bit is '0'. Where the constant c = 2 when Z' is DC coefficient, otherwise it set as 4.
Step 8 Repeat Steps 3-7 for remaining blocks until all watermark bits are extracted.
Step 9 Reconstruct watermark using these extracted watermark bits and the seed of PRNG.
After the action of Step 9 is finished, the watermark has been extracted to prove the user ownership.

2.4. **Lin et al. scheme [14].** In 2010, Lin et al. [14] presented a watermarking scheme against JPEG compression. Their scheme is a DCT-based image watermarking scheme, it hides the secret data at the low-frequency area and their watermark embedding and extracting procedures are described in details below. The watermark embedding procedure The original algorithm of Lin et al.'s scheme[14] is designed for color images, some steps of embedding and extracting phases use JPEG compression process to sample pixel value, transform DCT coefficients and quantize the related coefficients. Steps of Lin et al.'s watermark embedding procedure are given as follows. Step 1 Disarray watermark image by Torus Automorphism (TA) [18-20]. TA is a method to disarray watermark effectively [21], the formula of TA is given below:

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

FIGURE 1. JPEG standard quantization table of JPEG compression

$$\begin{pmatrix} i' \\ j' \end{pmatrix} = \begin{bmatrix} 1 & 1 \\ tt+1 \end{bmatrix} \times \begin{bmatrix} i \\ j \end{bmatrix} \bmod m, \tag{12}$$

Here i and j presents the original coordinate *(i, j)* of a watermark bit, respectively, and *i'* and *j'* presents the new coordinate *(i', j')* corresponding the original coordinate (i, j). And integer t and m are key parameters given by user.

Step 2 Obtain the luminance information Y of host color image by applying YUV color transformation. This step is used for transforming RGB color space into YUV color space. Then, sample the luminance information Y to embed watermarks. Lin et al. used YUV color space for watermarking instead of because RGB color space is highly related and is not proper for watermark embedding and only few schemes use blue channel for watermark embedding. As for using luminance information Y of host color image for watermark embedding, there two reasons: first, human visual system is much sensitive to the luminance information Y than other two chrominance components ($UandV$). Second, the luminance information Y has larger usage amount than other two chrominance components ($UandV$) in JPEG or MPEG applications. The JPEG standard's YUV color transformation formula is given below:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.148 & -0.289 & 0.437 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \tag{13}$$

In this transformation formula R, G and B means the red, green and blue pixel values of a color image, respectively. Y is the luminance component; and U and V are the chrominance components for the color image. After YUV transformation, the luminance Y plane is divided into non-overlapping blocks sized 8 8 and then used for hiding watermarks.

Step 3 Generate DCT coefficients by applying DCT transformation and JPEG standard quantization table shown in Fig. 1.

Step 4 Select the most complex blocks of image to embed watermark bits. In this step the more complex block means there are more non-zero coefficient in block.

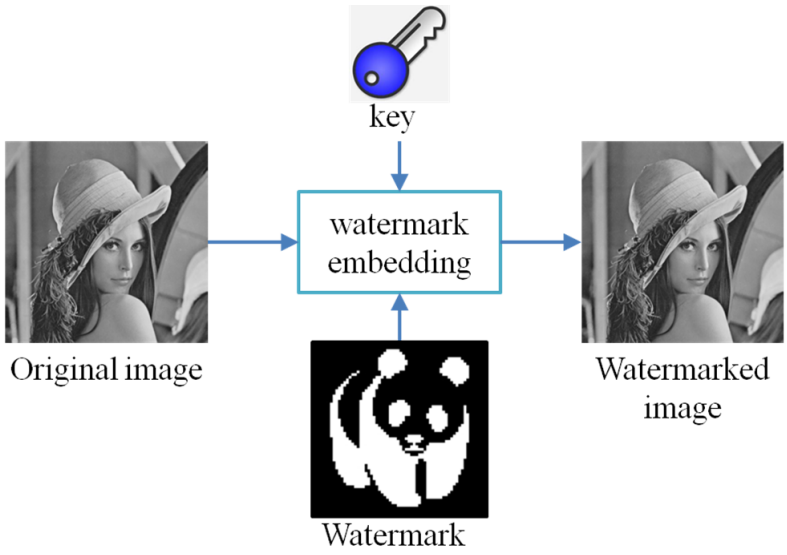

FIGURE 2. Framework of the watermark embedding process

Step 5 The DCT coefficients at low-frequency positions are selected to embedding watermark; each coefficient embeds one watermark bit. The selected coefficient is denoted as Z and let parameter M as the modulus. The relative formula is defined below.

$$\begin{aligned} Q &= \tfrac{|Z|}{M} \\ sign &= \begin{cases} 1, if Z \geq 0 \\ -1, if Z < 0 \end{cases} \end{aligned} \tag{14}$$

Step 6 Embed watermark bit into selected coefficient Z, then the watermarked coefficient Z' is obtained by following the rule. If watermark bit is equal to 0, the modify formula is defined as:

$$R = \frac{M}{4} \tag{15}$$

$$\begin{aligned} Z' &= sign \times ((Q) \times M + R) \\ Z' &= sign \times ((Q+1) \times M + R) \end{aligned} \tag{16}$$

If watermark bit is equal to 1 then the modify formula is defined as:

$$R = \frac{3M}{4} \tag{17}$$

$$\begin{aligned} Z'_{low} &= sign \times ((Q-1) \times M + R) \\ Z'_{high} &= sign \times ((Q) \times M + R) \end{aligned} \tag{18}$$

Finally, the Z' is determined according to the formula which is defined as:

$$Z' = \begin{cases} Z'_{low}, if\ |Z'_{low} - Z_{lin}| \leq |Z'_{high} - Z_{lin}| \\ Z'_{high}, if\ |Z'_{low} - Z_{lin}| > |Z'_{high} - Z_{lin}|' \end{cases} \tag{19}$$

Step 7 Repeat Steps 4-6 for remaining blocks until all watermark bits are embedded.
Step 8 Obtain the modified pixel blocks by applying inverse DCT transformation.

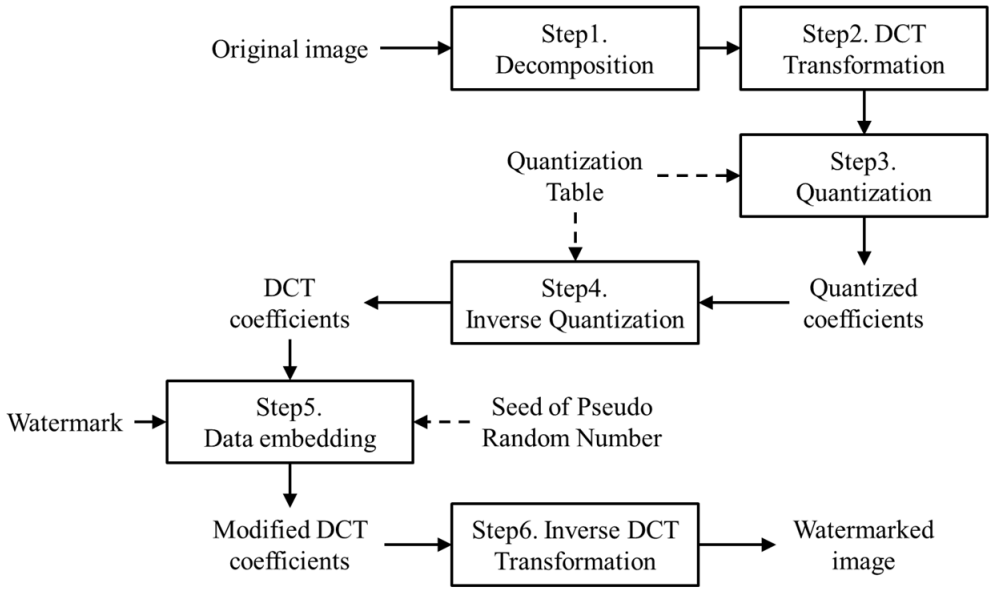

FIGURE 3. The system architecture diagram

2.4.1. *The watermark extracting procedure.* Before the watermark extracting process, four information must be obtained for watermark extraction, such as parameter t and m which used for TA permutation, the record of watermarked complex blocks, the position of watermarked coefficients and the modulus M. After obtain these information, follow these steps the embedded watermark could be reconstructed using for ensure ownership.
Step 1 Obtain the luminance information Y of watermarked image applying YUV color transformation shown in formula (13).
Step 2 Divide luminance Y into several non-overlapping 8 8 pixel blocks.
Step 3 Determine DCT coefficients applying DCT transform and quantization by quantization table.
Step 4 Extract the watermark bit form watermarked coefficient Z' by the formula which given below.

$$watermarkbit = \begin{cases} 0, if(|Z'| \bmod M) < \frac{M}{2} \\ 1, if(|Z'| \bmod M) \geq \frac{M'}{2} \end{cases} \tag{20}$$

Step 5 Repeat Steps 3-4 for remaining blocks until all watermark bits are extracted.
Step 6 Reconstruct watermark by disarray using Torus Automorphsim (TA).
When the action of step 6 is finished, the watermark would be obtained and used for providing the image copyright protection.

3. **The Proposed Scheme.** Here we present a robust watermarking scheme for image copyright protection. Fig. 2 shows the framework of this watermark embedding process. Only low frequency DCT coefficients are selected to carry a hidden watermark using the concept of mathematical remainder. We also use voting strategy to improve the robustness of the watermark against JPEG compression. Having explained our background logic, we move ahead to outline the principle of the proposed robust watermarking algorithm. The framework of our proposed watermark embedding process is shown in Fig. 2.

The details of our proposed watermark embedding process are shown in the following subsection.

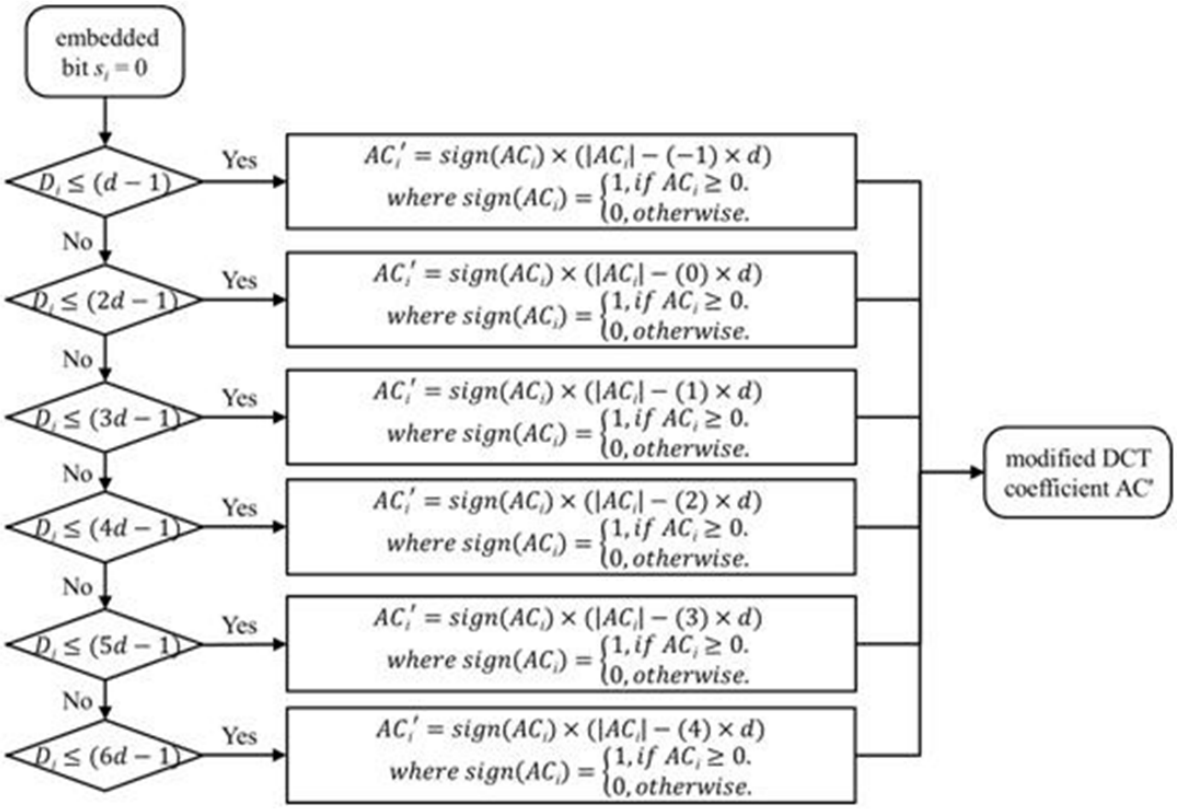

FIGURE 4. Embedding process for the watermarked bit "0"

3.1. **Watermark Embedding Procedure.** Fig. 3 illustrates the flowchart for the proposed watermark embedding process. The process begins by dividing the host image into 8×8-pixel blocks. We consider one block at a time to embed the watermark bits as shown in the following steps:
Step 1 Select an 8×8-pixel block from the host image. Step 2 Apply the DCT transformation to the selected 8×8 block. Step 3 Apply quantization to each block based on the quantization table provided by the JPEG compression standard. Step 4 Apply inverse quantization to each block based on the JPEG compression standard. Step 5 Randomly select a DCT coefficient AC_i to embed the watermark bit $s - i$ according to the pseudo random number generated by a predetermined secret key. Let M be the modulus, the relative variables can be computed as follows:

$$\begin{aligned} D_i &= |AC_i| \bmod M \\ d &= \left\lceil \tfrac{M}{6} \right\rceil \end{aligned} \tag{21}$$

where $D_i \in 0, 1, 2, \ldots, M-1$ is the mathematical remainder of $|AC_i|$, and d is the mathematical quotient obtained by dividing M by 6.
To embed watermark bit '0': $if (jd-1) < D_i \le \lceil (j+1)d - 1 \rceil, compute$:

$$AC'_i = sign(AC_i) \times (|AC_i| + (4+j) \times d), \tag{22}$$

where j= 0,1,2..,5. To embed watermark bit '1': $if (jd-1) < D_i \le \lceil (j+1)d - 1 \rceil, compute$

$$AC'_i = sign(AC_i) \times (|AC_i| + (4-j) \times d), \tag{23}$$

where j= 0,1,2..,5.

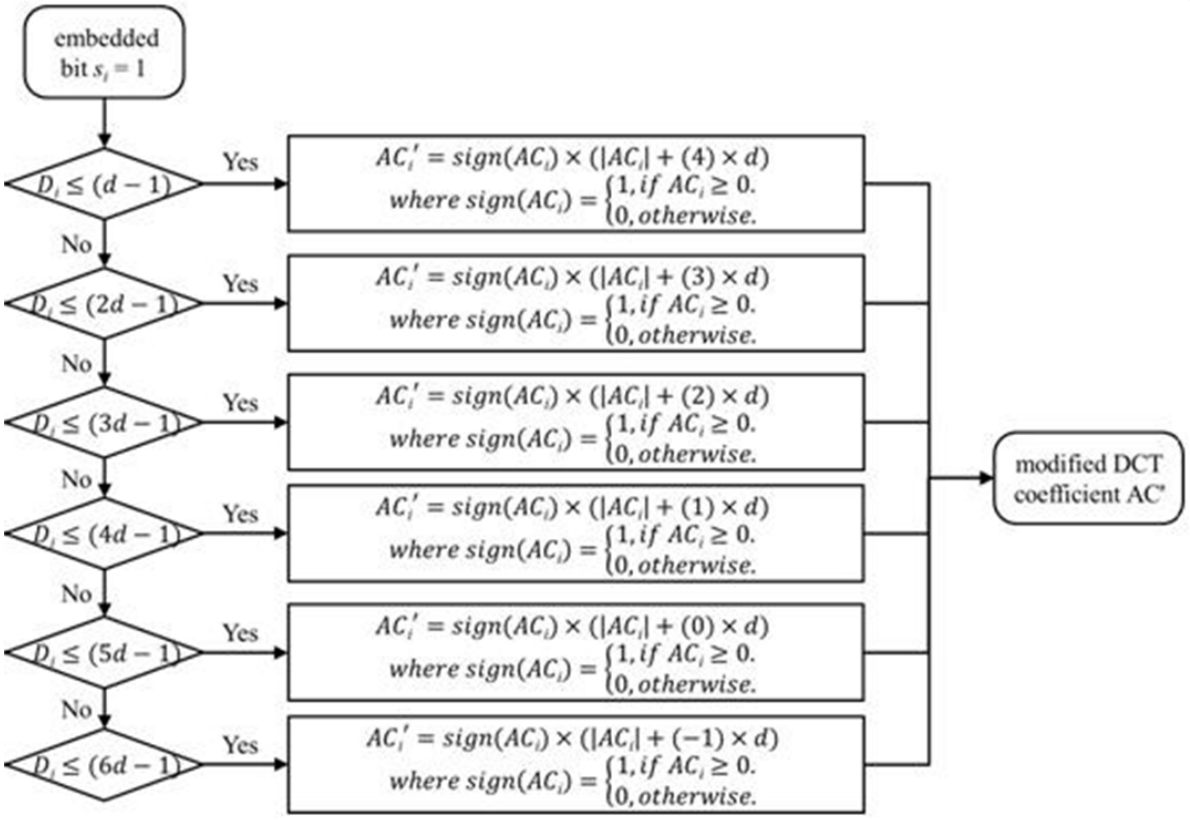

FIGURE 5. Embedding process for the watermarked bit "1"

Step 6 Apply an inverse DCT to the block to construct the watermarked image block. Fig. 4 and Fig. 5 show the embedding process for watermarked bit "0" and "1", respectively. Note that quantized coefficients may result in a wrong extraction of the watermark in Step 4. To avoid misjudgment of the extracted watermark, we use the offset variable, d, to provide a safe range for DCT coefficients. Fig. 6 presents the offset of the embedding process for $M = 18$. In the worst case of this watermarking embedding process, the largest embedding difference is $2M/3$. Cooperating with the watermark extracting algorithm, the proposed embedding process provides a safe range for the value of AC_i'. That is, even when the value of AC_i' is changed with a difference as large as M due to various attacks, the embedded watermark bit can be still extracted successfully.

3.2. **Watermarking Embedding Example.** Let AC_1, AC_2, AC_3, AC_4 be the four selected DCT coefficients for embedding, and $s = s_1, s_2, s_3, s_4 = 1, 0, 0, 1$ be the watermark. Fig. 7 shows an embedding example of four DCT coefficients for $M = 18$. Let us assume that the original DCT coefficients are $AC_1, AC_2, AC_3, AC_4 = -95, 47, -13, 74$. Since $D_1 = (-95) mod 18 = 5$, the first watermark bit 1 is embedded in AC_1 by setting $AC_1' = sign(AC_1) \times (|AC_1| + (3) \times d) = (-1) \times (|-95| + (3) \times 3) = -104$. The threshold $D_2 is 47 mod 18 = 11$, and the second watermark bit 0 is embedded in AC_2 by setting $AC_2' = sign(AC_2) \times (|AC_2| - (2) \times d) = (1) \times (|47| - (2) \times 3) = 41$. The watermark embedding process continues until all watermark bits are embedded, and the resulting watermarked DCT coefficients are obtained.

3.3. **Watermark Extracting Procedure.** Fig. 8 presents the flowchart for the watermark extraction process. The steps of the proposed watermark extraction algorithm are

$$D_i \equiv |AC_i| \ mod \ 18$$

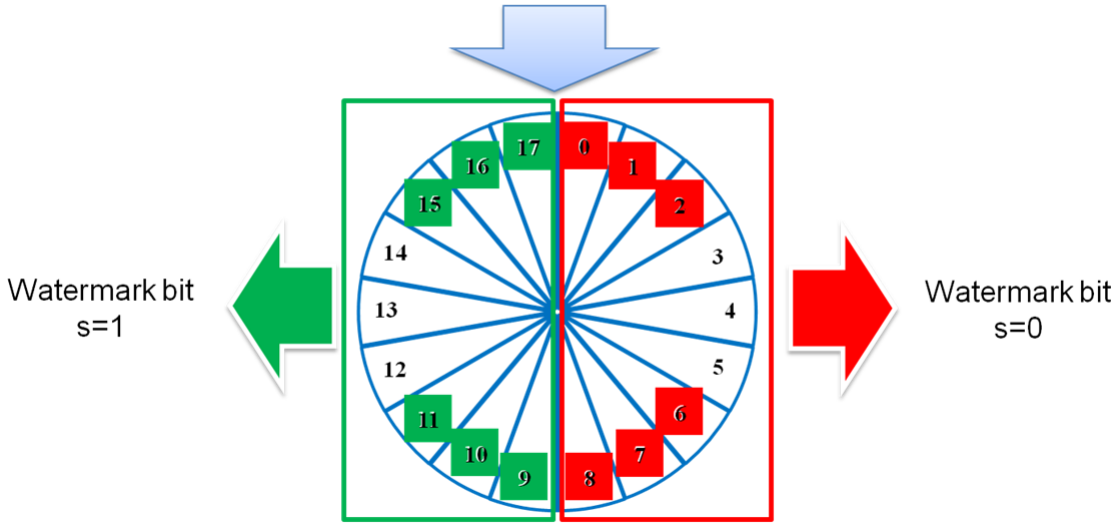

FIGURE 6. Offset of the embedding process for M=18

-95	47	-13	74
AC_1	AC_2	AC_3	AC_4

(a) Original DCT coefficients

1	0	0	1
s_1	s_2	s_3	s_4

(b) Watermark

5	11	13	2
D_1	D_2	D_3	D_4

(c) Threshold

-101	41	-4	71
AC_1'	AC_2'	AC_3'	AC_4'

(d) Watermarked coefficients

FIGURE 7. A watermark embedding example for four DCT coefficients for M=18

very similar to that for the watermark embedding algorithm. The procedure for watermark extraction is blind, that is, it does not need any assistance from the original image. Note that two key elements are needed in the watermark extraction procedure, the secret key that is predetermined during the watermark embedding process and used to generate a random number for embedding the watermark and the modulus M. The process begins by dividing the host image into 8×8-pixel blocks. The steps for watermark extraction are briefly listed below.

Step 1 Select an 8×8-pixel block from the host image.

Step 2 Apply DCT transform to the selected 8×8 block.

Step 3 Select a DCT coefficient AC_i' according to the pseudo random number generated by a predetermined secret key.

Step 4 Extract the watermark bit s_i' based on the following rule.

$$s' = \begin{cases} 0, if D'_i \leq \lceil \frac{M}{2} \rceil \\ 1, otherwise \end{cases} \tag{24}$$

where $D_i' = |AC_i'| mod M$.

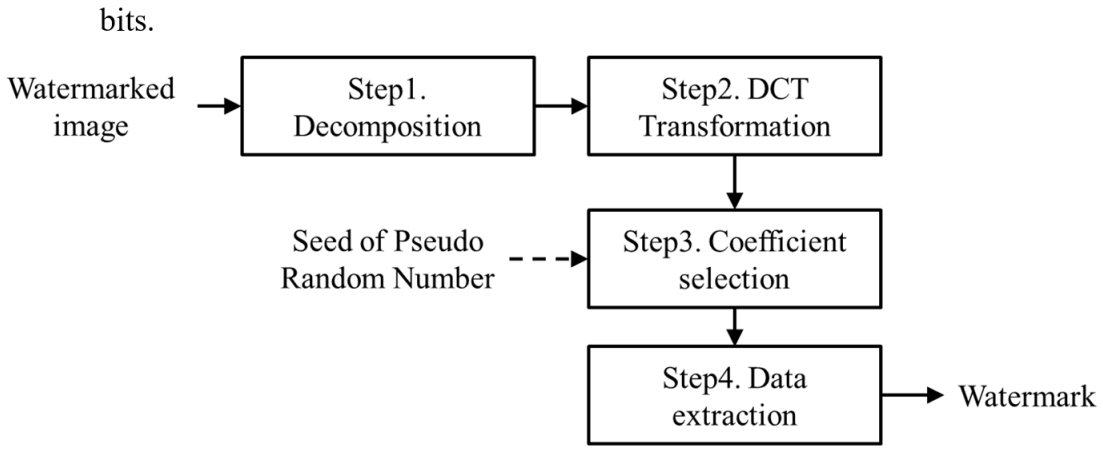

FIGURE 8. Flowchart of the watermark extraction process

FIGURE 9. Three copies of the watermark

FIGURE 10. 10 Binary watermark. (a) "Panda" of size 64×64. (b) "PU" of size 128×128

Step 5 Repeat Steps 1 to 4 for every consecutive block to extract all the watermark bits.

3.4. **Voting Strategy.** The extracted watermark bit s_i' should be identical to the original watermark bit si if no modifications have been made to the watermarked image. In other words, if the watermarked image has been attacked, the embedded watermark may not provide proof of ownership. Hence, we use a simple majority-voting strategy to improve the robustness of the hidden watermark.

TABLE 1. Hiding capacity vs. distortion for test images with M=48

PSNR	Lena	Mandrill	Boat	Jet	Pepper	Zelda
PSNR (dB) for 64×64 "Panda"	41.01	41.19	40.96	40.97	40.92	41.20
PSNR (dB) for 128×128 "PU"	33.73	34.87	33.74	33.76	33.87	33.79
PSNR (dB) for "Panda" with voting	36.22	36.65	36.10	36.09	36.22	36.16

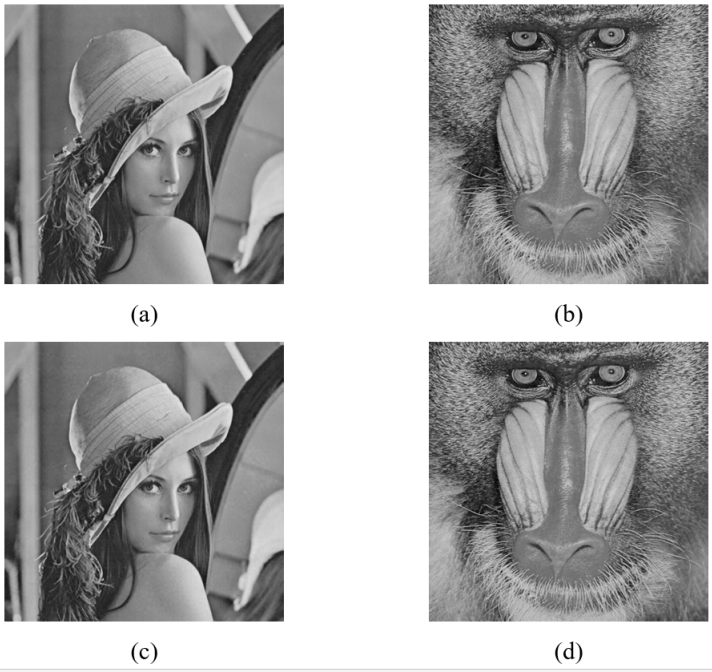

FIGURE 11. Watermarked "Lena" and "Mandrill" images. (a) 41.01 dB embedded with "Panda." (b) 41.19 dB embedded with "Panda." (c) 36.22 dB embedded with voting for "Panda." (d) 36.65 dB embedded with voting for "Panda".

Assume that the original watermark is the size 64×64 pixels. We first generate three copies of the watermark, as shown in Fig. 9. All three copies of the watermark are embedded in the original image using the proposed embedding algorithm. Then, we use a majority-voting strategy to reconstruct the original watermark.

$$W_o(x,y) = \{ \begin{matrix} W_{t2}(x,y), if W_{t2}(x,y) = W_{t3}(x,y), \\ W_{t1}(x,y), otherwise \end{matrix} \tag{25}$$

where $W_o(x,y), W_{t1}(x,y), W_{t2}(x,y)$, and $W_{t3}(x,y)$ represent the original watermark bit and the three copies at position (x, y), respectively.

4. **Experimental Results.** To obtain a clear understanding of how different attacks affect the performance of the proposed robust watermarking scheme, we present our results

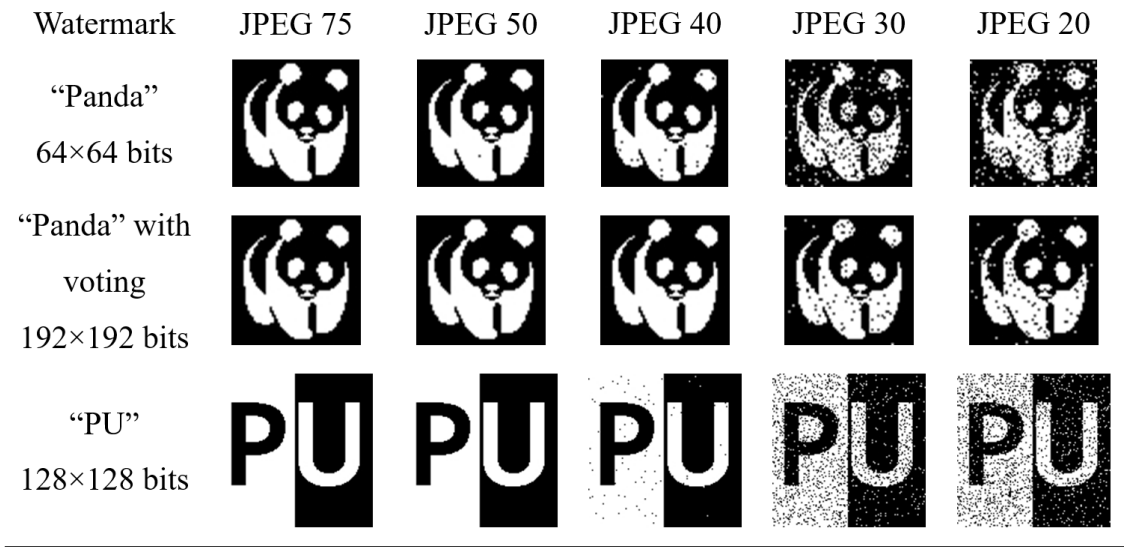

FIGURE 12. Extracted watermarks under JPEG compression taken from the watermarked image "Mandrill"

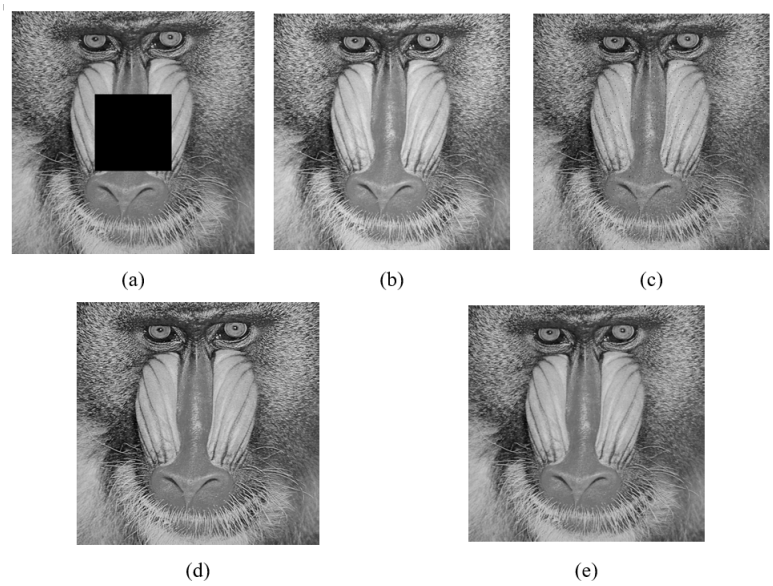

FIGURE 13. Watermarked "Mandrill" image under different attacks: (a) Cropping. (b) Brightening. (c) Addition of noise. (d) Sharpening. (e) JPEG compression

in a graphical form. All experiments were performed with six commonly used grayscale images sized 512×512, such as "Lena," "Mandrill," "Boat," "Jet," "Pepper," and "Zelda." The binary watermarks we tested in the experiments were "Panda" of size 64×64 and "PU" of size 128×128, as shown in Fig. 10.

To test fidelity and robustness performance, the two measures are defined below. Peak signal-to-noise ratio (PSNR) measures the fidelity, which refers to the perceptual quality of watermarked content.

$$\begin{gathered} PSNR(dB) = 10 \times \log_{10}\left(\frac{255}{MSE}\right)^2, \\ MSE = \frac{1}{H_o} \times \frac{1}{H_o} \times \left[\sum_{x=1}^{H_o} \sum_{y=1}^{W_o} \left(A(x,y) - \overline{A}(x,y)\right)^2\right] \end{gathered} \tag{26}$$

TABLE 2. PSNR and TAF obtained for various JPEG compression levels

Watermark	JPEG 75		JPEG 50		JPEG 40		JPEG 30		JPEG 20	
	PSNR (dB)	TAF (%)	PSNR (dB)	TAF (%)	PSNR (dB)	TAF (%)	PSNR (dB)	TAF (%)	PSNR (dB)	TAF (%)
"Panda"	32.63	0.00	29.71	0.12	28.82	0.56	27.80	8.08	26.49	8.64
"Panda" with voting	31.63	0.00	29.19	0.00	28.39	0.00	27.42	2.02	26.21	2.68
"PU"	30.98	0.00	28.82	0.00	28.07	0.50	27.14	8.85	26.04	10.11

MSE (Mean squared error) is a simple perceptual distance metric where A represents the host image, $\overline{A}$ represents the watermarked image, and $H_o \times W_o$ represents the size of the original and watermarked images. Tamper assessment function *(TAF)* [16, 17] is used to measure the extent of tampering of the extracted watermark. Considering the size of the watermark as $H_w \times W_w$, the TAF in percentage is defined as:

$$TAF(\%) = \frac{1}{H_w} \times \frac{1}{H_w} \times \left[\sum_{x=1}^{H_w} \sum_{y=1}^{W_w} \left(B(x,y) \oplus \overline{B}(x,y) \right)^2 \right] \times 100, \tag{27}$$

where $B(x,y)$ and $\overline{B}(x,y)$ represent the original and extracted watermarks at position, (x, y) respectively, and $\oplus$ is an exclusive-OR operator. The TAF represents the number of bits of the extracted watermark that are different from the original watermark, expressed in percent.

4.1. **Capacity vs. Distortion Performance.** Table I offers an example of how different watermark sizes influence the distortion for $M = 48$. Clearly, the PSNR abruptly decreases with increased watermark size. We also observed that images with abundant highly textured and noisy areas have generally higher PSNR values since the DCT coefficients of these images have a high variability. Further, even when we embed three copies of watermark using the proposed voting strategy, the PSNR is higher than 30 dB. As a result, our proposed scheme can provide an acceptable fidelity in exchange for higher robustness.

Fig. 11 shows the visual impacts of watermarked images at various hiding capacities for "Lena" and "Mandrill." In general, the watermarked image can hardly be distinguished from the original image. For the smooth image "Lena" while the voting strategy is used, the visual distortion is still quite small, and the PSNR is higher than 30 dB. It is well known that the human visual system (HVS) is less sensitive to errors for high frequency coefficients than it is to errors for lower frequency coefficients of DCT. Hence, images with high textured areas and low correlation, such as "Mandrill," embed more payload size at a higher PSNR.

4.2. **Robustness against JPEG Compression.** To test robustness against JPEG compression, the watermarked image was compressed by a JPEG algorithm with varying quality ranging from 50 to 75. The extracted watermarks under JPEG compression at varying levels are shown in Fig. 12. Using the proposed voting strategy, we can clearly recognize the extracted watermarks even when the compression quality is 20. It can be seen that there is a significant improvement in the quality of the hidden watermark extracted from the watermarked image subjected to JPEG compression.

The PSNR and TAF values of the extracted watermarks from "Mandrill" under different JPEG compression levels are summarized in Table II. Clearly, the watermark embedded using the proposed scheme can be almost fully extracted from the watermarked image when the compression quality is 40. In addition, generally the acceptance level of TAF is

TABLE 3. Performance comparison for the "Lena" image for our proposed scheme and three recent schemes [14, 16-17]

Scheme	Without attack		Brightening		Cropping		Sharpening		Noise		JPEG 75	
	PSNR	TAF	PSNR	TAF	PSNR	TAF	PSNR	TAF	PSNR	TAF	PSNR	TAF
Patra et al. scheme 1 [16]	61.09	0.00	29.46	44.79	15.60	5.17	28.82	40.21	25.32	0.61	39.87	43.48
Patra et al. scheme 2 [17]	40.45	3.24	29.17	14.57	15.58	7.95	28.50	13.30	25.16	21.11	37.18	5.05
Lin et al. scheme, M=26 [14]	39.05	0.00	29.08	0.39	15.58	5.00	28.16	39.47	25.24	21.65	36.49	0.00
Lin et al. scheme, M=48 [14]	33.12	0.00	28.02	0.00	15.53	5.05	26.79	44.77	24.85	15.28	32.30	0.00
Proposed scheme without voting, M=48	40.01	0.00	29.19	0.36	15.58	4.88	28.42	32.15	25.25	16.23	37.46	0.00
Proposed scheme with voting, M=48	36.22	0.00	28.68	0.00	15.56	0.00	27.77	28.95	25.25	8.59	34.67	0.00

TABLE 4. Performance comparison for the "Mandrill" image for our proposed scheme and three recent schemes [14, 16-17]

Scheme	Without attack		Brightening		Cropping		Sharpening		Noise		JPEG 75	
	PSNR	TAF	PSNR	TAF	PSNR	TAF	PSNR	TAF	PSNR	TAF	PSNR	TAF
Patra et al. scheme 1 [16]	61.39	0.00	29.39	43.75	14.26	5.17	18.56	45.36	25.78	0.46	33.26	42.99
Patra et al. scheme 2 [17]	39.33	2.73	28.99	13.96	14.25	7.29	18.53	27.95	25.33	20.99	32.32	4.39
Lin et al. scheme, M=26 [14]	40.60	0.00	29.43	0.43	14.26	24.3	18.52	45.70	25.47	22.58	32.54	0.00
Lin et al. scheme, M=48 [14]	34.46	0.00	28.34	0.02	14.22	5.29	18.43	43.26	25.15	18.84	30.82	0.00
Proposed scheme without voting, M=48	41.19	0.00	29.13	0.04	14.25	4.93	18.52	39.20	25.44	18.11	32.63	0.00
Proposed scheme with voting, M=48	36.65	0.00	28.67	0.00	14.24	0.00	18.50	34.42	25.42	9.15	31.63	0.00

15 % since the extracted watermark will not be recognizable above this value. As shown in Table II, the TAF value remains under 5 % for a JPEG compression quality from 20 to 75. As a result, the proposed scheme is quite robust for varying levels of JPEG compression.

4.3. **Comparison to Other Recent Schemes.** Table III compares the ability among our proposed scheme and other recent schemes [14, 16-17] under different types of attacks in image quality for "Lena." The watermarked image was subjected to the following five different attacks:

1. Cropping of a 10% block size in the middle of the watermarked image
2. Brightening the watermarked image to 110%
3. Adding noise to the entire watermarked image with a 25% distortion rate
4. Sharpening the watermarked image by 50%
5. JPEG compression with a compression quality of 75.

Samples of the watermarked images "Mandrill" under the same attacks stated above are shown in Fig. 13. As Table III indicates, the schemes proposed by Patra et al. [16] provide best performance of PSNRs. However, their scheme is not able to withstand general attacks except noise attack. Since their scheme produces a quite high TAF value which causes the extracted watermark to become unrecognizable. They further improved the weakness of their early works [16] and proposed the improved version later. In the improved scheme, Patra et al.'s scheme [17] still has some problems such the extracted watermark is not always the same as the original one even without any attack. Therefore, Lin et al. [14] and our two proposed schemes present better performance in the robustness than Patra et al.'s scheme [16-17]. Moreover, it can be seen from Table III that there is substantial improvement of our proposed schemes in the brightening and cropping performance over which schemes [14, 16-17]. For sharpening and noise attack, the TAF performance of the proposed schemes is similar to that of schemes [16-17] but superior than Lin et al.'s scheme [14]. For whole common attacks, the proposed scheme maintained the lowest TAF value. In addition, the performance comparison for the high textured image

TABLE 5. Performance comparison for JPEG compression for the "Lena" image with schemes [14, 16-17]

Scheme	JPEG 50		JPEG 40		JPEG 30		JPEG 20	
	PSNR	TAF	PSNR	TAF	PSNR	TAF	PSNR	TAF
Patra et al.'s scheme 1 [16]	37.54	43.04	36.74	44.26	35.70	44.26	34.13	46.06
Patra et al.'s scheme 2 [17]	35.79	9.69	35.32	10.47	34.59	16.08	33.38	19.99
Lin et al.'s scheme, *M*=26 [14]	35.31	10.54	34.79	21.85	33.98	13.91	33.24	30.12
Lin et al.'s scheme, *M*=48 [14]	31.86	0.00	31.35	0.00	30.89	5.81	30.92	24.70
Proposed scheme without voting, *M*=48	35.95	0.00	35.31	0.17	34.53	8.37	33.39	5.78
Proposed scheme with voting, *M*=48	33.79	0.00	33.28	0.00	32.81	4.19	32.15	1.14

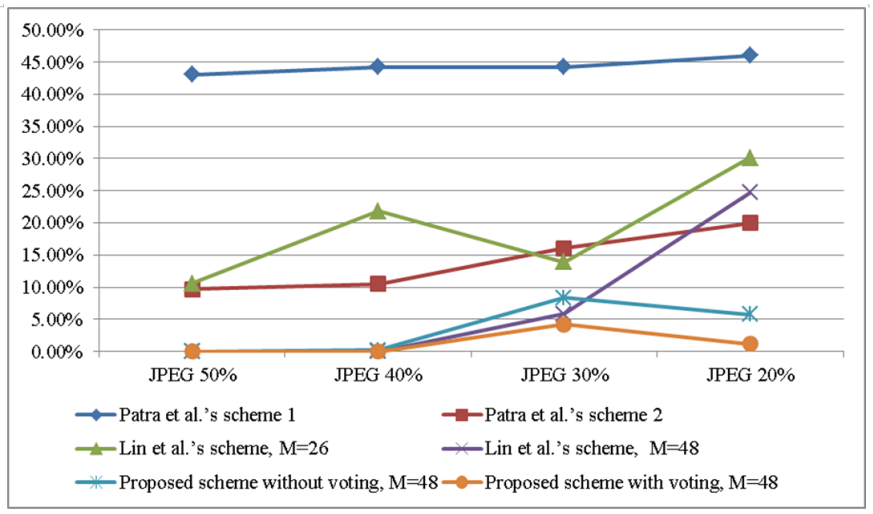

FIGURE 14. . TAF of extracted watermark under JPEG compression for schemes [14, 16-17] for "Lena"

"Mandrill" can be seen in Table IV. In the case of high textured images, the proposed scheme maintains its superiority over schemes [14, 16-17].

To show the improvement of the proposed scheme in terms of JPEG compression, we compared our scheme both voting version and non-voting version to schemes [14, 16-17], and that performance comparison is given in Table V. The watermarked image was compressed by JPEG algorithm with varying quality ranging from 20 to 50. As seen from Table V, the proposed scheme both voting version and non-voting version are quite robust for varying levels of JPEG compression: the TAF value remained under 6% for JPEG compression quality from 20 to 50. In contrast, scheme [16] was not able to withstand any JPEG compression since it produces quite high TAF value which causes the extracted watermark to be unrecognizable. Though scheme [17] has improvement the robust of scheme [16], its TAF value still higher than proposed scheme non-voting version. On the other hand, Lin et al.'s scheme [14] with $M = 48$, it remained the TAF value under 24% for JPEG compression quality from 20 to 50. Though Lin et al.'s scheme [14] has the better robustness than Patra et al.'s schemes [16-17], our scheme with voting version still has the best robust. Fig. 14 demonstrates that the proposed scheme achieves good improvement for robustness against JPEG compression. These experimental results illustrate that a noticeable improvement in robustness, both for the common attacks and JPEG compression, is achieved by the proposed scheme.

5. **Conclusions.** In this paper, we presented a DCT-based, robust watermarking scheme for proof of ownership. We use the concept of mathematical remainder to modify the DCT coefficient to ensure its robustness against incidental attacks and malicious attacks. The circle property generated by the module operation provides a better robustness for JPEG attacks. A majority-voting strategy is applied to the scheme to enhance the robustness of the watermark. The voting strategy embedded three copies of watermark into the watermarked image. Each watermark would be distributed into image randomly. When the watermarked coefficient has broken, it could get the correct watermark bit form the other coefficients by voting strategy. Though this idea would decrease PSNR value, the experiment results appear that watermark robust has significant improvement. With the proposed scheme, the embedded watermark can successfully survive after being attacked by image processing operations, especially for the JPEG compression with various compression levels. The simulation results show that the proposed scheme outperforms the earlier work. As a result, our proposed scheme is more suitable for the JPEG image that is the most common graphics format found on the Internet.

REFERENCES

[1] I. J. Cox, M. Miller, J. Bloom, and M. Miller, *Digital watermarking, Morgan Kaufmann Publishers, Inc., San Francisco*, 2001.

[2] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, Secure spread spectrum watermarking for multimedia*IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1673-1687, 1997.

[3] F. Liu and C. K. Wu, Robust visual cryptography-based watermarking scheme for multiple cover images and multiple owners*IET Information Security*, vol. 5, no. 2, pp. 121-128, 2011.

[4] S. C. Pei, J. M. Guo, and H. Lee, Novel robust watermarking technique in dithering halftone images*IEEE Signal Processing Letters*, vol. 12, no. 4, pp. 333-336, 2005.

[5] C. C. Lin, W. L. Tai, and C. C. Chang, Multilevel reversible data hiding based on histogram modification of difference images*Pattern Recognition*, vol. 41, no. 12, pp. 3582-3591, 2008.

[6] W. L. Tai, C. M. Yeh, and C. C. Chang, Reversible data hiding based on histogram modification of pixel differences*IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 6, pp. 906-910, Jun. 2009.

[7] X. Zhang, S. Wang, Z. Q, and G. F, Reference sharing mechanism for watermark self-embedding*IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 485-495, 2011.

[8] C. C. Lin and P. F. Shiu, DCT-based reversible data hiding scheme*Journal of Software,* vol. 5, no. 2, pp. 214-224, 2010.

[9] C. C. Chang, C. C. Lin, C. S. Tseng, and W. L. Tai, Reversible hiding in DCT-based compressed images*Information Sciences*, vol. 177, no. 13, pp. 2768-2786, 2007.

[10] K. Maeno, Q. Sun, S. F. Chang, and M. Suto, New semi-fragile image authentication watermarking techniques using random bias and non-uniform quantization*IEEE Transactions on Multimedia*, vol. 8, no. 1, pp. 32-45, 2006.

[11] J. Huang, Y. Q. Shi, and Y. Shi, Embedding image watermarks in DC components*IEEE Transactions on Circuits and Systems for Video Technology,* vol. 10, no. 6, pp. 974-979, Sep. 2000.

[12] S. D. Lin and C. F. Chen, A robust DCT-based watermarking for copyright protection*IEEE Transactions on Consumer Electronics*, vol. 46, no. 3, pp. 415-421, 2000.

[13] S. D. Lin, S. C. Shie, and C. F. Chen, A DCT based image watermarking with threshold embedding*International Journal of Computers and Applications*, vol. 25, no. 2, pp. 130-135, 2003.

[14] S. D. Lin, S. C. Shie, and J. Y. Guo, Improving the robustness of DCT-based image watermarking against JPEG compression*Computer Standards & Interfaces*, vol. 32, no. 1-2, pp. 54-60, Jan. 2010.

[15] Y. H. Ku, X. Sun, The Chinese remainder theorem*Journal of The Franklin Institute*, vol. 329, no. 1, pp. 93-97, 1992.

[16] J. C. Patra, A. Karthik, and C. Bornand, A novel CRT-based watermarking technique for authentication of multimedia contents*Digital Signal Processing*, vol. 20, no. 2, pp. 442-453, Mar. 2010.

[17] J. C. Patra, J. E. Phua, and C. Bornand, A novel DCT domain CRT-based watermarking scheme for image authentication surviving JPEG compression*Digital Signal Processing,* vol. 20, no. 6, pp. 1597-1611, Dec. 2010.

[18] I. Percival and F. Vivaldi, Arithmetical properties of strongly chaotic motions*Physica D Nonlinear Phenomena*, vol. 25, pp. 105-130, 1987.

[19] D. K. Arrowsmith and C. M. Place, An introduction to dynamical systems, , Cambridge Univ., Press, 1990.

[20] G. Voyatzis, and I. Pitas, Chaotic mixing of digital images and applications to watermarking*Proceedings of the European Conference on Multimedia Applications, Services and Techniques*, vol. 2, pp. 687-695, 1996.

[21] C. C. Chang, J.Y. Hsiao, and C.L. Chiang, An image copyright protection scheme based on torus automorphism*Proceedings of the First International Symposium on Cyber Worlds*, pp. 217-224, 2002.

[22] C. C. Chang, T. S. Nguyen, M.-C. Lin, C.-C. Lin, A novel data-hiding and compression scheme based on block classification of SMVQ indices*Digital Signal Processing,* vol. 51, pp. 142-155, 2016.