Journal of Information Hiding and Multimedia Signal Processing ©2010 ISSN 2073-4212
Ubiquitous International Volume **1**, Number **3**, July **2010**

High Capacity Data Hiding Scheme for DCT-based Images

Chia-Chen Lin

Department of Computer Science and Information Management
Providence University
200 Chung-Chi Rd., Taichung, Taiwan
mhlin3@pu.edu.tw

Pei-Feng Shiu

Department of Computer Science and Information Engineering
Providence University
200 Chung-Chi Rd., Taichung, Taiwan
g9872001@pu.edu.tw

Received May 2009; revised May 2010

Abstract. *To enhance the hiding capacity of DCT-based images, in this paper we propose a new data hiding scheme based on a notation transformation concept. Without considering the reversibility function, experimental results confirm that the hiding capacity provided by our designed data hiding strategy is not only higher than that provided by the Iwata et al.'s, Chang et al.'s and Lin et al.'s schemes but also Lin and Shiu's scheme. Moreover, the image quality of stego-images with our proposed scheme remains above 30 dB for most test images when the hiding capacity is above 90000 bits, which is better than the best image quality offered by existing DCT-based loss or lossless data hiding schemes. Our proposed scheme also inherits the advantage of DCT-based data hiding scheme; that is the security of the hidden data is guaranteed because the secret data is embedded into DCT coefficients of the cover image. Based on the results of our proposed scheme, our future work is to add the reversibility function back into the data hiding scheme to expand the applications of our proposed scheme.*

Keywords: High hiding capacity, data hiding, DCT coefficients, JPEG

1. **Introduction.** Protecting data transmitted over the Internet has become a critical issue driven by the progress in data digitalization and communications networking techniques over the past decade. To ensure transmitted data are secure and cannot be tampered with or eavesdropped upon by malicious attackers, scholars have proposed two approaches over the past ten years. One is a traditional cryptographic approach, in which secret messages are transformed into unrecognizable form by using secret information, also called keys, which are shared between senders and authorized receivers. In this approach, only authorized users can retransform a meaningless secret message back to its original form. Many well-known cryptographic schemes, such as RSA [18], DES [9], are widely used in the commercial market.

The other approach is steganography, which enables senders to transmit secret messages via meaningful cover media, such as video, audio, images or text documents, to avoid attracting attackers' attention. Hiding the subjects with steganographic techniques using

meaningful images involves the spatial [1, 4, 13, 16-17, 19, 21], compression [5, 6, 7, 20] and frequency [2-3, 10, 12-15, 22] domains of those cover images.

In the spatial domain approach, the secret message is embedded directly into the pixels of a cover image. Least significant bit (LSB)-based hiding strategies are most commonly used in this approach. For example, in Lee and Chen's scheme [16], the LSB of each pixel in a cover image is modified to hide the secret message. In Chang et al.'s scheme [4], a dynamic programming strategy is used to find the optimal LSB substitution in order to hide images. In addition to LSB-based hiding strategies, several schemes that use different strategies to hide secret messages in the spatial domains of cover images also have been proposed [16, 17, 19]. For example, Chung et al. offered the singular value decomposition (SVD)-based hiding scheme [7], Tsai et al. used the bit plane of each block truncation coding (BTC) block to embed secret messages [20], and Chang et al. used the GA algorithm and absolute moment BTC to embed secret messages into color images [6].

In the frequency domain [2-3, 10, 12-15, 22], cover images must first be transformed using a frequency-oriented mechanism such as discrete cosine transform (DCT), discrete wavelet transform (DWT) or similar mechanisms, after which the secret messages can be combined with the coefficients in the frequency-form images to achieve embedding. For example, in Chang et al.'s scheme [3], the medium-frequency coefficients of DCT-transformed cover images are used to embed a secret message. The JPEG quantization table is also modified to further protect the embedded secret message. Similarly, Iwata et al. use the boundaries between zero and non-zero DCT coefficients to hide secret data [12]. In 2007, Chang et al. extended Iwata et al.'s idea [12] and presented a lossless steganographic scheme for hiding secret data in each block of quantized DCT coefficients in JPEG images [2]. In Chang et al.'s scheme, the two successive zero coefficients of the medium-frequency components in each block are used to hide secret data. They further modified the quantization table to maintain the quality of the stego-image while concealing a higher payload compared with Iwata et al. scheme. Thus, their scheme achieves reversibility and acceptable image quality of the stego-image simultaneously. However, their scheme can only embed secret bits into the zero coefficient located in the successive zero coefficients in the medium area; non-zero coefficients in the medium area cannot be used. Later, Lin et al. embedded the secret values into the middle-frequency of the quantized DCT coefficients, and limit the number of nonzero values of the quantized DCT coefficients which participate in the data hiding procedure to design a adaptive and reversible DCT-based data hiding scheme [15]. In 2009, Lin and Shiu combined Chang et al.'s [2] scheme and then designed a 2-layers data hiding scheme for DCT-based images. Lin and Shiu's [14] scheme outperforms Chang et al.'s scheme [2] in hiding capacity but the size of the hidden secret data is still less than 70000 bits on average because it retains the reversibility function. To enhance the hiding capacity of DCT-based images and explore the relationship between hiding capacity and image quality, in this paper we propose a high-capacity data hiding scheme for DCT-based images based on a notational transformation concept but without the reversibility function. Experimental results confirm that the proposed scheme successfully enhances hiding capacity while maintaining acceptable image quality.

The rest of this paper is organized as follows. In Section 2, we briefly review the DCT transform, and three selected schemes: Iwata et al.'s data hiding scheme [12], Chang et al.'s data hiding scheme [2] and Lin and Shiu's [14] scheme. Our proposed high-capacity data hiding scheme is demonstrated in Section 3, and Section 4 presents our experimental results. Finally, concluding remarks appear in Section 5.

2. **Related Work.** In this section, we briefly review the DCT transform and then introduce three selected DCT-based data hiding schemes: Iwata et al.'s data hiding scheme [12], Chang et al.'s [2] and Lin and Shiu's [14] reversible data hiding schemes.

2.1. **Discrete Cosine Transform (DCT) and Quantization.** DCT is a widely used mechanism for image transformation adopted to compress JPEG images [11]. Figure 1 shows the JPEG compression process, which consists of five phases: transforming an RGB image to a YCbCr image, composition of minimum coding units, 2-dimensional DCT, quantization of DCT coefficients, runlength coding and Huffman coding.

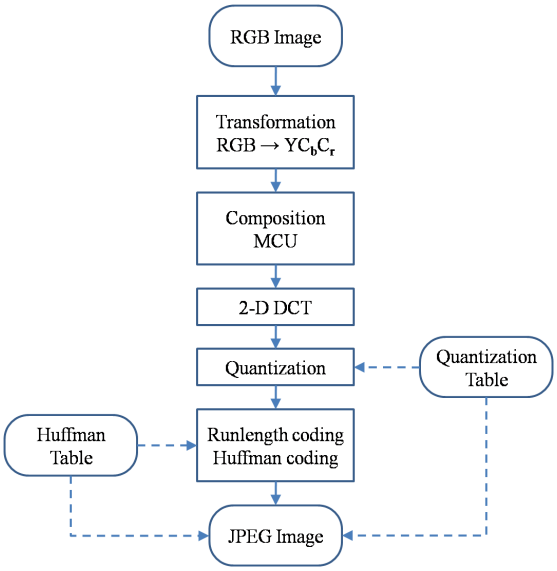


FIGURE 1. Flowchart of JPEG compression

In the 2-dimensional DCT phase, each 8×8 non-overlapping block is transformed into the DCT domain by using the 2-dimensional DCT in Equation (1).

$$F(u,v) = \frac{c(u)c(v)}{4} \sum_{i=0}^{7} \sum_{j=0}^{7} \cos(\frac{(2i+1)u\pi}{16}) \cos(\frac{(2j+1)v\pi}{16}) f(i,j),$$
$$\text{where} c(e) = \begin{cases} \frac{1}{\sqrt{2}}, & \text{if } e = 0. \\ 1, & \text{if } e \neq 0. \end{cases} \quad (1)$$

Here, F(u,v) and f(i,j) present a DCT coefficient at the (u,v) coordinate and a pixel value at the (i,j) coordinate, respectively. F(0,0) is the DC component, which corresponds to an average intensity value of each block in the spatial domain. F(u,v) is the AC component, in which $u \neq 0$ and $v \neq 0$. For data reduction during the quantization phase, DCT coefficients are quantized by using the standard quantization table shown in Figure 2.

The human vision system is much more sensitive to the values in low-frequency components than those in the higher frequencies. Thus, distortion in high-frequency components is visually acceptable and perceptible. Therefore, the upper left values in the quantization table in Figure 2 are small enough to avoid large alteration. In contrast, the lower right values in the table are large and can be altered.

2.2. **Iwata et al.'s Data hiding Scheme.** In 2004, Iwata et al. discovered that the values of AC coefficients tend to be zero after the quantization phase of JPEG compression; therefore, they designed a data hiding strategy that embeds secret information into

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 2. Standard quantization table

high-frequency components based on the length of zero sequences after quantization of the DCT coefficients [12]. By using Iwata et al.'s modification strategy, a JPEG-coded stego-image can be obtained after runlength coding and Huffman coding. The modified DCT coefficients are reserved after the secret data are extracted from the JPEG-coded stego-image because both runlength coding and Huffman coding involve lossless compression. Iwata et al.'s data hiding scheme is divided into two phases: data hiding and data extracting. Note that Iwata et al.'s scheme requires a preprocessing phase because the secret data are hidden in the DCT coefficients of a cover image. The preprocessing phase in Iwata et al.'s scheme involves first partitioning a cover image into non-overlapping blocks of 8×8 pixels, then performing the 2-dimensional DCT to transform each block into an 8×8 block of DCT coefficients. Later, the quantized coefficients are obtained through the 8×8 quantization table in Figure 2. After the DCT coefficients are quantized, Iwata et al.'s data hiding phase can begin. The following subsections contain detailed descriptions of the data hiding and data extracting phases, respectively.

2.2.1. *Data Hiding Phase.* First, Iwata et al. defined a set for embedding 1 bit for data hiding, as shown in Figure 3.

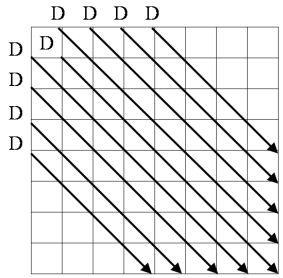


FIGURE 3. Sets for Iwata et al.'s data sets for data hiding

In Figure 3, set D_i $(1 \leq i \leq l_i)$ contains quantized DCT coefficients on the line labeled "D_i". Let l_i be the length of a zero sequence of higher frequency components on the "D_i" line. The odd or even state of l_i indicates what secret data are embedded into D_i. Let $(d_{i,1},d_{i,2},d_{i,3},...,d_{i,k})$ be the coefficient sequence in set Di with k_i components from low frequency to high frequency, $d_{i,j}$ be the non-zero value of the highest frequency component of set Di where $1 \leq j \leq k_i$, and T be a predetermined threshold. Based on these sets of a block, Iwata et al. designed four modification strategies for four cases. Basically, they modified the length of the zero sequence of higher frequency components on line "D_i" to let l_i be even when a secret bit is 1, and odd when a secret bit is odd. The detailed modification strategies for four cases are described below.

Case 1: When $|d_{i,j}| > T$ the coefficient of the location $d_{i,j+1}$ is replaced by 1 or -1, where

-1 and 1 are randomly selected.
Case 2: When $|d_{i,j}| \leq T$ and $d_{i,j-1}$ is equal to 0, $d_{i,j}$ and $d_{i,j-1}$ will be replaced with 0 and 1, respectively.
Case 3: When $|d_{i,j}| \leq T$ but $d_{i,j-1}$ does not exist or is not equal to 0, $d_{i,j}$ will be set at 0.
Case 4: When $d_{i,j}$ does not exist, all components in set D_i are zero, and 1 or -1 is randomly assigned to the lowest coefficient in the set D_i.

2.2.2. *Data Extracting Phase.* After the receiver receives a JPEG stego-image from a sender, s/he conducts the following steps to extract the hidden secret information from each 8×8 block in a JPEG stego-image:
Step 1. Obtain 8×8 non-overlapping blocks of the quantized DCT coefficients of the Y component from a JPEG stego-image after Huffman decoding and runlength decoding.
Step 2. Extract secret bits from an 8×8 non-overlapping block by using Equation (2).

$$w_i = \begin{cases} 0, & \text{when } l_i \text{ is even.} \\ 1, & \text{when } l_i \text{ is odd.} \end{cases} \tag{2}$$

Here, w_i $(1 \leq i \leq l_i)$ is the hidden bit in the set D_i.

By using this data hiding strategy, Iwata et al. successfully embedded a secret bit into a set in an 8×8 block of a JPEG image and caused only a slight difference in the histogram of quantized DCT coefficients. Later, Chang et al. [2] found the weakness of Iwata et al.'s scheme; that is, the lack of reversibility. To overcome this weakness, Chang et al. proposed reversible data hiding for JPEG-coded images in 2007. A detailed description of Chang et al.'s proposed scheme is presented in subsection 2.3.

2.3. **Chang et al.'s Reversible Data Hiding Scheme.** As mentioned in the previous subsection, Chang et al.'s scheme successfully embeds secret bits into a DCT-based compressed image and restores the original DCT coefficients after the secret bits are extracted [2], which cannot be achieved with Iwata et al.'s scheme [12]. Chang et al.'s scheme can be divided into three phases: data hiding, data extracting and restoring. Note that their scheme also requires a preprocessing phase because the secret data are hidden in the DCT coefficients of a cover image. The preprocessing phase in the proposed scheme involves first partitioning a cover image into non-overlapping blocks of 8×8 pixels, then performing the 2-dimensional DCT to transform each block into an 8×8 block of DCT coefficients. Later, the quantized coefficients are obtained through the 8×8 quantization table in Figure 2. After the DCT coefficients are quantized, Chang et al.'s data hiding phase can begin.

2.3.1. *Data Hiding Phase.* Chang et al.'s scheme defines several sets of R_i $(1 \leq i \leq 9)$ for embedding one data bit, as shown in Figure 4. In each set, one secret bit is embedded into the successive zero sequence, which runs from the highest frequency component to the lower frequency components and ensures at least two zeros in each set R_i $(1 \leq i \leq 9)$. Let b_i $(1 \leq i \leq 9)$ be the length of ceaseless zeros in order from the highest frequency component to the lower frequency components in set R_i. The value of b_i is key to deciding whether set R_i can carry a secret bit in Chang et al.'s scheme. The estimation rule is as follows: if $b_i \geq 2$, set R_i can hide a secret bit; otherwise, set R_i cannot hide a secret bit. In the example shown in Figure 5, four continuous zero sequences exist from the highest frequency component to the lower frequency components in set R_1, and b_1 equals 4 because the length of ceaseless zeros in order from the highest frequency component

to the lower frequency components in set R_1.is 4. Similarly, $b_2 = 2$ and $b_3 = 1$ can be obtained, respectively.

Following these rules, in set R_i, if $b_i \geq 2$, $z_{i,1}$ represents the zero value of the lowest frequency of set R_i, and $z_{i,2}$ represents the lower right component of $z_{i,1}$. Note that $z_{i,2}$ does not exist once b_i is less than 2 in set R_i (e.g., in the set R_3 shown in Figure 6). Let be the coefficient sequence in set R_i with k_i components from high frequency to low frequency, and let s_i be the secret bit to be embedded in set R_i. Refer to set R_1 in Figure 5. In set R_1, the coefficient sequence is represented as $(r_{i,1}, r_{i,2}, r_{i,3}, r_{i,4}, r_{i,5}, r_{i,6}, r_{i,7})$, and the values of set R_1 are (0, 0, 0, 0, 2, 2, 3), individually. According to the definition just given, $z_{1,1}$ stands for $r_{1,4} = 0$ and $z_{1,2}$ stands for $r_{1,3}$ in set R_1.

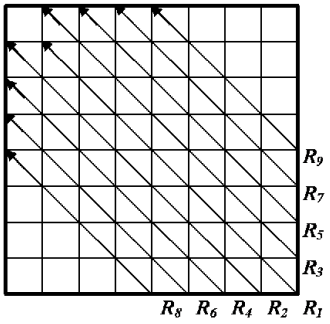


FIGURE 4. Coefficient sets for embedding

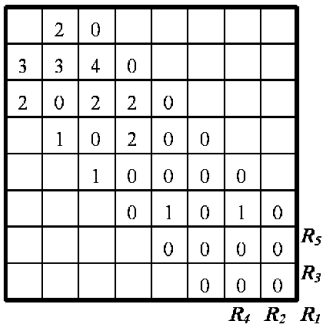


FIGURE 5. Example of quantized coefficients

To make sure the embedded coefficients can be successfully restored during the restoration phase, Chang et al.'s data hiding strategies and embedding rules for ambiguous conditions are listed below.

Rule 1: If $b_i \geq 2$, the value of $z_{i,2}$ is used to indicate the hidden secret bit in set R_i $(1 \leq i \leq 9)$. The value of $z_{i,2}$ is modified to hide the secret bit by using Equation (3).

$$z_{i,2} = \begin{cases} 0, & \text{when } s_i \text{ is 0.} \\ 1 \text{ or } -1, & \text{when } s_i \text{ is 1.} \end{cases} \tag{3}$$

Note that 1 or -1 is randomly selected to $z_{i,2}$ when s_i is 1.

Ambiguous Condition A and its remedial solution: Before data hiding can take place, Chang et al.'s scheme need to eliminate any potentially ambiguous conditions. If the sequence of set R_i is $(0, 0, K, x, 0)$ and all coefficients of $(r_{i,1}, r_{i,2}, ..., r_{i,j-2})$ are zeros,

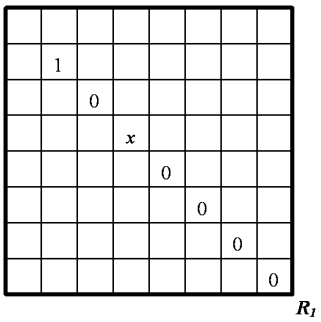


FIGURE 6. Example of an ambiguous condition when x =1 or -1

where $x \neq 0$, $4 \leq j \leq k_i$. According to this definition, $z_{i,2}$ is $r_{i,j-3}$ in set R_i. Once secret bit s_i equals 1 and x is 1 or -1, the receiver might make a false judgment while extracting data from the set R_i. Figure 6 demonstrates an ambiguous condition for x=1 or -1 and the coefficients located in the components at frequencies higher than x are all zeros. In addition, the value of the upper left component of x is also zero.

To avoid this ambiguous condition and guarantee that the original coefficient can be successfully restored, the coefficient $r_{i,j-1}$ is modified as shown in Equation (4) before the secret bit can be hidden.

$$r'_{i,j-1} = \begin{cases} r_{i,j-1} + 1, & \text{when } r_{i,j-1} > 0. \\ r_{i,j-1} - 1, & \text{when } r_{i,j-1} < 0. \end{cases} \text{ where } 3 \leq (j-1) \leq k_i, \tag{4}$$

Return to set R2 in Figure 5. The corresponding coefficient sequence ($r_{2,1}$, $r_{2,2}$, $r_{2,3}$, $r_{2,4}$, $r_{2,5}$, $r_{2,6}$, $r_{2,7}$) of set R_2 is (0, 0, 1, 0, 0, 0, 3). In set R_2, $r_{2,3}$ is $z_{i,2}$, so $r_{2,3}$ must be modified to hide the secret bit. However, once $r_{2,3}$ is modified according to Equation (3), the receiver may make the misjudgment that the hidden bit is $r_{2,3}$ instead of $r_{2,1}$. To avoid this potential misjudgment, the value of $r_{2,3}$ must be changed from 1 to 2 according to Equation (4). The modified coefficient sequence of set R_2 is then presented as (0, 0, 2, 0, 0, 0, 3). In general, the successful embedding of each set of a DCT-quantized coefficient block should cause no more than two coefficients to be modified.

Rule 2: If $b_i < 2$ and both $z_{i,1}$ and $z_{i,2}$ do not exist, none secret bits can be hidden in a set R_i.

Two ambiguous conditions may exist; therefore, two remedial measures for eliminating such conditions are described below.

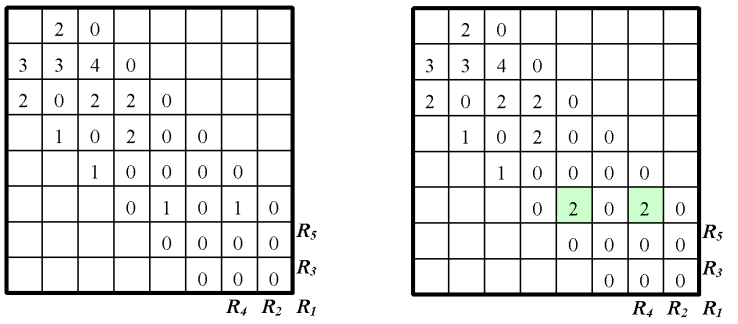
Ambiguous Condition B and its remedial solution: If the two highest coefficients $r_{i,1}$ and $r_{i,2}$ of set R_i are x and 0, respectively, $r_{i,j}$ is changed according to Equation (5) to demonstrate no secret data are hidden and make sure the ambiguous condition is not repeated.

$$r'_{i,1} = \begin{cases} r_{i,1} + 1, & \text{when } r_{i,1} > 0. \\ r_{i,1} - 1, & \text{when } r_{i,1} < 0. \end{cases} \tag{5}$$

Ambiguous Condition C and its remedial solution: If the three highest coefficients $r_{i,1}$,$r_{i,2}$ and $r_{i,3}$ of set R_i are 0, x and 0, respectively, the value of $r_{i,2}$ is modified according to Equation (6).

$$r'_{i,2} = \begin{cases} r_{i,2} + 1, & \text{when } r_{i,2} > 0. \\ r_{i,2} - 1, & \text{when } r_{i,2} < 0. \end{cases} \tag{6}$$

Example of Chang et al.'s Data Hiding Strategy: The example in Figure 7 clearly demonstrates the Chang et al.'s data hiding strategy. Note that R_2 and R_3 in Figure 7(a) are the hidden coefficients without ambiguous conditions, but R_1, R_4 and R_5 show ambiguous conditions after data hiding, which may lead to the restoration phase not working properly. Therefore, the elimination solutions must apply to these three sets. Figure 7(b) shows the embedded coefficients without ambiguous conditions when Chang et al.'s data hiding strategies are used. The eliminated results are shown in Table 1.



(a) Original DCT coefficient in a block

(b) Modified coefficient block without an ambiguous situation

FIGURE 7. Example of hiding four bits into five sets in a block

TABLE 1. Sets of Figure 7(b), secret bits and final hidden results

Set	k_i	Coefficients $(r_{i,1} \to r_{i,k})$	$z_{i,2}$	s_i	Mofified coefficients $(r'_{i,1} \to r'_{i,k})$
R_1	7	(0,0,0,0,2,2,3)	$r_{1,3}$	0	(0,0,**0**,0,2,2,3)
R_2	7	(0,0,2,0,0,0,3)	$r_{2,1}$	0	(**0**,0,2,0,0,0,3)
R_3	7	(0,2,0,0,2,4,2)	not exist	not exist	(0,2,0,0,2,4,2)
R_4	6	(0,0,0,0,1,1,2)	$r_{4,2}$	1	(0,**1**,0,1,1,3)
R_5	6	(0,0,0,0,0,0,0)	$r_{5,5}$	1	(0,0,0,0,**1**,0)

2.3.2. *Data Extracting Phase.* Chang et al.'s data extracting phase consists of five steps, detailed descriptions of which follow.
Step 1. Obtain non-overlapping 8×8 blocks of the quantized DCT coefficients of the Y components from a JPEG stego-image after Huffman decoding and runlength decoding.
Step 2. Scan each block according to a predetermined order.
Step 3. For each set R_i in a block, let $r_{i,j}$ be the highest frequency non-zero component, where $1 \leq i \leq 9$ and $1 \leq j \leq k_i$.
Step 4. Extract si from set Ri by using the following rules:

Rule 1: If $r_{i,j} = 1$ or -1 and $r_{i,j+1} = 0$, then s_i is 1 and mark $r_{i,j}$ as $z_{i,2}$.
Rule 2: If $r_{i,j} = 1$ or -1, $r_{i,j+1} \neq 0$, $r_{i,j-1} = 0$ and $r_{i,j-2} = 0$, then s_i is 0 and mark $r_{i,j-2}$ as $z_{i,2}$ where $j - 2 \geq 1$.
Rule 3: If $r_{i,j} = 1$ or -1 and $r_{i,j+1} \neq 0$ for $j \leq 2$, none secret bit in set R_i. That is, s_i does not exist in set R_i.
Rule 4: If $r_{i,j} \neq 1$ or -1, $r_{i,j-1} = 0$ and $r_{i,j-2} = 0$, then s_i is 0 and mark $r_{i,j-2}$ as $z_{i,2}$,

where $j-2 \geq 1$.
Rule 5: If $r_{i,j}$ 1 or -1 and $j \leq 2$, none secret bits are in set R_i. That is, s_i does not exist in set R_i.
Rule 6: If $r_{i,j}$ does not exist, then s_i is 0 and mark $r_{i,1}$ as $z_{i,2}$.

Step 5. Repeat Steps 3 and 4 until all blocks are processed.

Take Table 2 for example. In set R_1, the highest frequency non-zero value is $r_{1,5}$ and the pair ($r'_{1,3}$,$r'_{1,4}$) is (0, 0), which satisfies Rule 4, so secret bit s_1 is 0. The secret data in set R_2 are extracted in the same way as R1 and secret bit s_2 is 0. No secret bit is hidden in set R_3 because $r'_{3,1}$ does not equal 1 or -1 and Rule 5 is satisfied. In set R_4, the highest frequency non-zero coefficient is $r'_{4,2}$ and equals 1. Moreover, the value of $r'_{4,3}$ is 0, which satisfies Rule 1; therefore, the secret bit is 1. Based on the same rule, the secret bit extracted from set R_5 is also the same as in R_4.

TABLE 2. Sets of final hidden result, extracted secret bits and related data

Set	k_i	Mofified coefficients ($r'_{i,1} \rightarrow r'_{i,k}$)	$z_{i,2}$	s_i
R_1	7	(0,0,0,0,2,2,3)	$r'_{1,3}$	0
R_2	7	(0,0,2,0,0,0,3)	$r'_{2,1}$	0
R_3	7	(0,2,0,0,2,4,2)	not exist	not exist
R_4	6	(0,1,0,0,1,1,2)	$r'_{4,2}$	1
R_5	6	(0,0,0,0,0,1,0)	$r'_{5,5}$	1

2.3.3. *Data Restoring Phase.* Because Chang et al.'s scheme offers reversibility, the restoring phase begins when the data extraction phase is completed. As Table 2 shows, some sets may not contain any secret bits because their b_i's are less than 2. During the data extraction phase, their scheme has already recognized the corresponding z_i,2 for each embeddable set. In the data restoring phase, the proposed scheme must first replace z_i,2 in each embeddable set with 0. Later, the original value of the modified coefficient in each set can be restored based on the three rules below. Let the location of z_i,2 in set R_i be $r'_{r,j}$ in each embeddable set.
Rule 1: If s_i exists and $r_{i,j+3} = 0$, where $4 \leq (j+3) \leq k$, then the original value of $r_{i,j+2}$ is restored by using Equation (7).

$$r_{i,j+2} = \begin{cases} r'_{i,j+2} - 1, & \text{when } r'_{i,j+2} > 0, \\ r'_{i,j+2} + 1, & \text{when } r'_{i,j+2} < 0, \end{cases} \quad \text{where } 3 \leq (j+2) < k_i. \tag{7}$$

Rule 2: If s_i does not exist and the two highest coefficients $(r'_{i,1}, r'_{i,2})$ of set R_i equal (x, 0), where $x \neq 0$, then the original value of $r'_{i,1}$ is restored by using Equation (8).

$$r_{i,1} = \begin{cases} r'_{i,1} - 1, & \text{when } r'_{i,1} > 0, \\ r'_{i,1} + 1, & \text{when } r'_{i,1} < 0. \end{cases} \tag{8}$$

Rule 3: If s_i does not exist and the pair having the three highest coefficients $(r'_{i,1}, r'_{i,2}, r'_{i,3})$ of set R_i equal (0, x, 0), where $x \neq 0$, then the original value of $r'_{i,2}$ is restored by using Equation (9).

$$r_{i,2} = \begin{cases} r'_{i,2} - 1, & \text{when } r'_{i,2} > 0, \\ r'_{i,2} + 1, & \text{when } r'_{i,2} < 0. \end{cases} \tag{9}$$

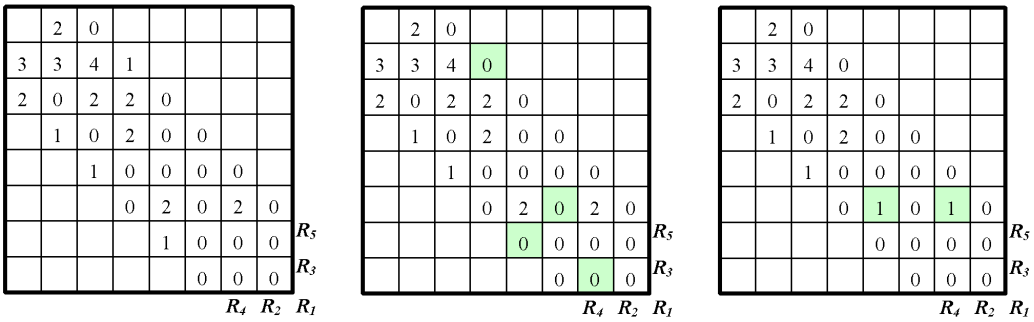


FIGURE 8. Example of Chang et al.'s process for restoring DCT coefficients

Figure 8(a) presents the final hidden results and Figure 8(b) presents the results after $z_{i,2}$ is replaced with 0 in each embeddable set. Take set R_2 as an example. During the data extraction phase, $r'_{2,1}$ is recognized as $z_{2,2}$ in set R_2 in Table 2. In set R_2, the coefficient sequence is (0, 0, 2, 0, 0, 0, 3) running from high frequency to low frequency, as shown in Figure 8(b). Because a secret bit exists and $r_{2,4}$ is 0, which satisfies Rule 1 for restoring the original coefficient, $r_{2,3}$ is changed to 1 with Equation (6). Set R_3 is not an embeddable set but its three highest coefficients are (0, 2, 0), which satisfies Rule 3; therefore, $r_{3,2}$ is also replaced with 1 by using Equation (8). The restored DCT coefficient block is shown in Figure 8(c). Note that Figure 8(c) is exactly the same as Figure 7(a).

2.4. **Lin et al.'s Reversible Data Hiding Scheme.** Inspired by Chang et al.'s idea, Lin and Shiu proposed a 2-layers data hiding scheme for DCT-based images in 2009 [14]. The following subsections introduce Lin and Shiu's reversible data hiding scheme.

2.4.1. *Data Hiding Phase.* Basically, Lin and Shiu's scheme [14] is an extension of Chang et al.'s scheme; therefore, their proposed embedding phase combines two data hiding strategies. One is Chang et al.'s data hiding strategy, called layer-2 data hiding, and the other is Lin and Shiu's proposed data hiding strategy, called layer-1 data hiding. Because Chang et al.'s data hiding strategy [2] was introduced in subsection 2.2, in this subsection we focus on Lin and Shiu's proposed data hiding strategy.

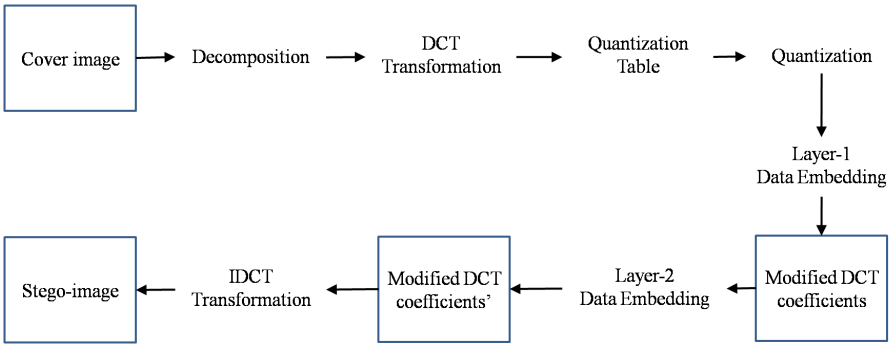


FIGURE 9. Flowchart of Lin and Shiu's reversible data hiding

The basic idea behind Lin and Shiu's data hiding strategy is to hide one secret bit into two neighboring coefficients in a set. To implement such a data hiding mechanism, Lin

and Shiu adopted Tian's pixel expansion method [19]. Moreover, an indicator is used in Lin and Shiu's scheme to distinguish a coefficient as the hidden result when its value is "0". Three equations from Tian's scheme were used to design their data hiding strategy in layer-1 data hiding, as follows:

$$avg = \lfloor \frac{P1 + P2}{2} \rfloor, \tag{10}$$

$$d = P1 - P2 \text{ and } d' = 2 \times d + s, \tag{11}$$

$$P1' = avg + \lfloor \frac{d' + 1}{2} \rfloor \text{ and } P2' = avg - \lfloor \frac{d'}{2} \rfloor. \tag{12}$$

To make sure the hidden secret bit can be extracted during the extracting phase and the two embedded neighboring coefficients can be successfully restored in the restoring phase, we set the embedding conduction as follows: in a set, a non-zero DCT coefficient and a zero DCT coefficient (x, 0) are selected as the embedding pair when they are next to successive zero coefficients. Take R_1 :(0, 0, 0, 0, 2, 2, 3) shown in Figure 7(a) for example. In R_1, (2,0) is an embeddable pair when the coefficient pattern is either N N x 0 0 0 or N N 0 x 0 0 where N denotes an integer and x denotes any integer except 0 and -1. Once an embeddable pair is determined, Equations (10) to (12) are used to hide a secret bit s in the embedding pair. For each embeddable pair, the next zero coefficient must be modified so that misjudgment does not occur during the data extraction phase.

To avoid an ambiguous condition occurring after data hiding, Lin and Shiu designed four ambiguous solutions in their proposed layer-1 data hiding strategy, as defined in the following rules.

Rule 1: If an embeddable pair (x, 0, 0) is changed to (y, 0, 0) where x and y are non-zero coefficients that do not equal -1 after data hiding, the next zero coefficient is changed to "-1". The embedded pair (y, 0, 0) is changed to (y, 0, -1).

Rule 2: If an embeddable pair (x, 0, 0) is changed to (0, y, 0) where x and y are non-zero coefficients that do not equal -1 after data hiding, the next zero coefficient is changed to "-1". The embedded pair (0, y, 0) is changed to (0, y, -1).

Rule 3: If an embeddable pair (x, 0, 0) is changed to (y, -1, 0) where x and y are non-zero coefficients that do not equal -1 after data hiding, the next zero coefficient is changed to "-1". The embedded pair (y, -1, 0) is changed to (y, -1, -1).

Rule 4: The unembeddable pair (N, N, -1) is changed to (N, N, 1) to make sure misjudgment does not occur during the data extraction phase.

2.4.2. *Data Extracting Phase.* Because Lin and Shiu used Tian's pixel expansion method to design their embedding strategy for two neighboring coefficients (x, 0) that are close to successive zero coefficients, Tian's data extraction phase can also be used to extract the hidden secret bit from an embedded pair. Tian's data extraction phase can be easily implemented by using Equations (13) to (15) in combination with Lin and Shiu's six data extraction rules, which are defined below.

The corresponding extraction equations are shown below.

$$avg = \lfloor \frac{P1' + P2'}{2} \rfloor, \tag{13}$$

$$s = d' - \lfloor \frac{d'}{2} \rfloor \times 2 \text{ and } d = \lfloor \frac{d'}{2} \rfloor, \tag{14}$$

$$P1 = avg + \lfloor \frac{d + 1}{2} \rfloor \text{ and } P2 = avg - \lfloor \frac{d}{2} \rfloor. \tag{15}$$

Rule 1: If the coefficient pattern (N, N, N, x, 0, -1) is found, where N and x denote an integer coefficient and any non-zero coefficient except -1, respectively, Equations (13) to (15) are used to extract a hidden secret bit from (x, 0) in the embedded pair during the data extracting phase.
Rule 2: If the coefficient pattern (N, N, N, x, -1, -1) is found, where N and x denote an integer coefficient and any non-zero coefficient except -1, respectively, Equations (13) to (15) are used to extract a hidden secret bit from (x, -1) in the embedded pair during the data extracting phase.
Rule 3: If the coefficient pattern (N, N, N, 0, x, -1) is found, where N and x denote an integer coefficient and any non-zero coefficient except -1, respectively, Equations (13) to (15) are used to extract a hidden secret bit from (0, x) in the embedded pair during the data extracting phase.
Rule 4: If the coefficient pattern (N, N, N, x, y, 0) is found, where N denotes an integer coefficient and x and y denote any non-zero coefficient except -1, Equations (13) to (15) are used to extract a hidden secret bit from (x, y) in the embedded pair during the data extracting phase.
Rule 5: If the coefficient pattern (N, N, N, 0, y, 0) is found, where N and y denote an integer coefficient and any non-zero coefficient except -1, respectively, no secret is contained in the coefficient pattern during the data extracting phase.
Rule 6: If the coefficient pattern (N, N, N, N, N, y) is found, where N and y denote an integer coefficient and any non-zero coefficient except -1, respectively, no secret is contained in the coefficient pattern during the data extracting phase.

2.4.3. *Data Restoring Phase.* Once the conditions in Rules 1 to 3 in the data extracting phase are met, the data restoring phase can begin. Basically, three data restoring cases must be considered. The corresponding restoring process for each data restoring case is listed below.
Case 1: If the embedded coefficient pattern (N, N, N, x, 0, -1) is found, where N and x denote an integer coefficient and any non-zero coefficient except -1, respectively, during the data extracting phase, the indicator "-1" is changed to "0" in the coefficient pattern during the data restoring phase.
Case 2: If the embedded coefficient pattern (N, N, N, x, -1, -1) is found, where N and x denote an integer coefficient and any non-zero coefficient except -1, respectively, during the data extracting phase, the indicator "-1" is changed to "0" in the coefficient pattern during the data restoring phase.
Case 3: If the embedded coefficient pattern (N, N, N, 0, x, -1) is found, where N and x denote an integer coefficient and any non-zero coefficient except -1, respectively, during the data extracting phase, the indicator "-1" is changed to "0" in the coefficient pattern during the data restoring phase.

In addition, if the coefficient pattern (N, N, N, x, y, 1) is found, where N denotes an integer coefficient and x and y denote any non-zero coefficient except -1 during the data extracting phase, change (N, N, N, x, y, 1) to (N, N, N, x, y, 0) during the data restoring phase.

3. **Out Proposed High-Capacity Data Hiding Scheme.** As stated at the beginning of this paper, we discovered that Chang et al.'s [2] and Lin and Shiu's [14] schemes achieve reversibility and maintain acceptable image quality at the same time, but neither effectively uses the coefficients located in the medial frequency because its hiding strategies modify only the zero coefficient located in successive zero coefficients. Such hiding strategies could offer reversibility at the cost of lower hiding capacity.

To overcome the limitations of hiding capacity of DCT-based images, we explore the upper bound of hiding capacity of DCT-based images when reversibility is not concerned. The following subsections give a detailed description of our proposed high-capacity data hiding scheme.

3.1. **Data Hiding Phase.** To explore the upper bound of hiding capacity of a DCT-based image, we first divide the coefficients in the middle frequency of a DCT-based image into the six sub-areas shown in Figure 10.

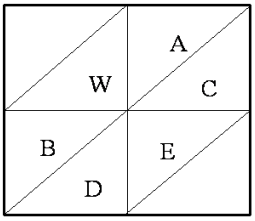


FIGURE 10. Six sub-areas of coefficients located in the middle frequency of a DCT-based image

Next, we design a hiding strategy based on a notational transformation concept. A flowchart of our proposed scheme is shown in Figure 11.

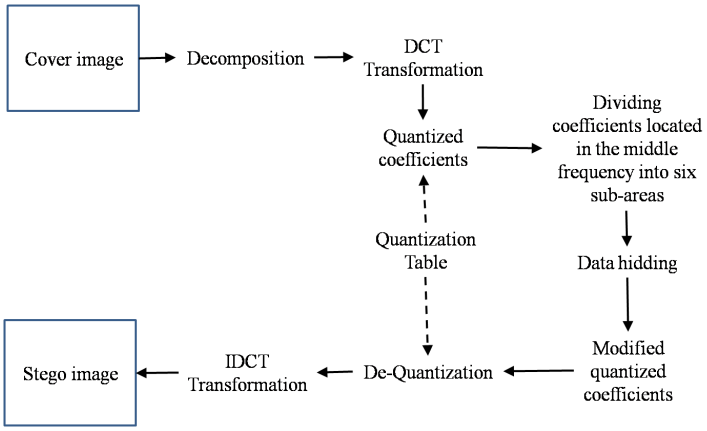


FIGURE 11. Flowchart of our proposed data hiding phase

The most significant difference between the proposed data hiding scheme and previous DCT-based data hiding schemes is the data hiding strategy; therefore, we only demonstrate our proposed data hiding strategies in detail as follows.
Step 1. Transform the secret stream into 3^2 system; for example, $s = (110100)^2$ the transformed secret bitstream is represented as $\delta = (57)_9$.
Step 2. Select two neighboring coefficients as a pair denoted as (X, Y).
Step 3. Transform two neighboring coefficients into an integer α by using Equation (16).

$$\alpha = (3^0 \times X + 3^1 \times Y) \bmod 3^2, \tag{16}$$

Step 4. Compute the difference μ between the integer αrepresenting the coefficient pair and the value δ of the transformed secret bitstream by using Equation (17).

$$\mu = (\delta_i - \alpha + \lfloor \frac{3^2 - 1}{2} \rfloor) \bmod 3^2, \tag{17}$$

Where αrepresents the coefficient pair and the value δ_i represents the ith integer value of the transformed secret bitstream δ. Take $\delta = (57)_9$ for example: δ_1 is the 1th integer value "5" and δ_2 is the 2th integer.
Step 5. Transform the difference μ into a stream μ^T by using a 3-based notation transformation, where μ^T is denoted as $\mu^T = (a_1, a_2)$.
Step 6. Each element in μ^T is decreased by 1 and the decreased stream is denoted as $\overline{\mu^T} = (a_1, a_2)$.
Step 7. Switch elements in $\overline{\mu^T}$ so that $\overline{\mu^T} = (a_2, a_1)$.
Step 8. Modify the coefficients in the coefficient pair by using Equation (18) to generate the hidden coefficients.

$$X' = X + a_2 \text{ and } Y' = Y + a_1, \tag{18}$$

To demonstrate our proposed data hiding strategy, we assume two neighboring coefficient pairs as (4, 6) and the 3^2-based transformed secret stream as $\delta = (5)_9$. By using Equation (16), we derive α=4=$((3^0 \times 4 + 3^1 \times 6) \bmod 3^2)$. By using Equation (17), μ is computed as μ=5=((5-4 + (9-1)/2) mod 9) = ((1 + 4) mod 9). μ^T is obtained as (1, 2) after transforming μ into a stream by using a 3-based notation transformation and $\overline{\mu^T} = (0, 1)$ after Step 6 is performed. After Step 7 is performed, the elements are switched to (1, 0). After using Equation (18), the modified pixels concealing the secret data are generated as (4, 6)=(5, 6).

Some rules must be followed during the data hiding phase to make sure the hidden secret data can be extracted during the data extraction phase. These special cases are listed below:
Rule 1: For two kinds of coefficients pairs (0, 0) or (1, 1), the coefficients must be randomly modified as (1, -1) or (-1, 1) to make sure the hidden data can be successfully extracted during the data extraction phase.
Rule 2: Even when the difference μ is equal to or larger than 9, the number of its transform must remain as 2. For example, when μ=9 its μ^T =(3, 0), and when μ=17 its μ^T =(5, 2).
Rule 3: Even when $\delta_i - \alpha + \lfloor \frac{3^2-1}{2} \rfloor$ is less than 0, μ is always a positive integer after the mod operation is performed. For example, when $(\delta_i - \alpha + \lfloor \frac{3^2-1}{2} \rfloor)$=-1, its μ=8=$((\delta_i - \alpha + \lfloor \frac{3^2-1}{2} \rfloor) \bmod 3^2)$ rather than -1.

3.2. **Data Extracting Phase.** After the receiver receives the stego-image, the decoder performs the same DCT transformation and quantization as in the data hiding phase. Next, the decoder divides the coefficients in the middle frequency of the DCT-based stego-image and picks two neighboring coefficients to extract the hidden data according to Equation (19). Let us assume that the two neighboring hidden coefficients are denoted as (X', Y').

$$\delta_i = (3^0 \times X' + 3^1 \times Y') \bmod 3^2, \tag{19}$$

where the value δ_i represents the ith integer value of transformed secret bitstream δ. This operation is repeated until all coefficients are processed. Finally, the transformed secret bitstream δ is transformed into a binary system. For example, $\delta = (57)_9$ is transformed

into $s = (110100)_2$ together instead of each integer in the transformed secret bitstream δ being transformed into the 2-based notation system individually.

Following the example given in the data hiding phase, the same example in Table 3 is used to demonstrate our proposed data extraction phase.

TABLE 3. Hidden coefficients generated by our proposed data hiding strategy

Integers in transformed secret stream δ	Original coefficients	Hidden coefficients
$\delta 1$=5	(4, 6)	(5, 6)
$\delta 2$=7	(7, 2)	(7, 3)

In Table 3, the hidden coefficients are (5, 6) so that the hidden secret data are $\delta_1 = ((3^0 \times 5 + 3^1 \times 6) \bmod 3^2)$=5. As for the hidden coefficients (7, 3), the hidden secret data are $\delta_2 = ((3^0 \times 7 + 3^1 \times 3) \bmod 3^2)$=7. After all hidden secret data are extracted, the secret data are transformed into a 2-based notation system at the same time as $(57)_9 = (110100)_2$.

4. **Experiments.** To prove that the hiding capacity of our proposed data hiding scheme overcomes the limitations for a DCT-based image while maintaining acceptable image quality of the stego-images, in this section we further discuss the hiding capacity and image quality of stego-images. Our data hiding and extracting algorithms were developed using C programming language. Our simulation platform is Microsoft Windows XP, Pentium 4 CPU with 2GB memory.

(a) Boats (b) F16 (c) Lena

(d) Baboon (e) Peppers (f) Zelda

FIGURE 12. Six 512 × 512 gray-level test images

In our first experiment, we used the six 512×512 gray-scale images shown in Figure 12 as our cover images, to which we applied the following algorithm.
Input: A 512×512 gray-scale image as a cover image.
Output: A compressed image.
Step 1: Divide the cover image into several non-overlapping blocks.
Step 2: Use DCT to transform each 8×8 block into the DCT coefficients for each block.
Setp 3: Perform quantization with the standard quantization table and our modified quantization table as shown in Figures 2 and 13.
Step 4: Use an inverse DCT process to transform each block into the spatial domain.

Although the cover images used in our scheme are only achieved by quantization, they remain the same as those generated by JPEG compression because quantization is the only lossy process in JPEG compression. Therefore, the following experimental results are very similar to those achieved using JPEG compression images as cover images.

16	11	10	16	24	40	51	61
12	12	14	19	26	41	60	55
14	13	16	17	28	40	48	56
14	17	22	20	36	61	56	43
18	22	26	39	48	76	72	54
24	25	39	45	57	73	79	64
49	64	55	61	72	85	84	71
72	92	95	69	78	70	72	70

FIGURE 13. Modified quantization table

The peak signal to noise ratio (*PSNR*) used to evaluate image quality is defined in Equation (20).

$$PSNR = 10 \log_{10} \frac{255^2}{MSE}, \tag{20}$$

where the mean square error (*MSE*) for an $M \times N$ gray-level image is defined in Equation (21).

$$MSE = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} x_{i,j} - x'_{i,j}{}^2, \tag{21}$$

where $x_{i,j}$ and $x'_{i,j}$ are the pixel values of the cover images and stego-images, respectively.

When comparing the six stego-images in Figures 14 and 15, it becomes obvious that the *PSNR*s of reconstructed images generated by the modified quantization table are higher that those generated by the standard quantization table. Therefore, the modified quantization table is used to work with our proposed data hiding scheme, the generated stego-images and their corresponding *PSNR*s are shown in Figure 16. After comparing the image quality of the reconstructed images shown in Figures 14 to 16, we found that the image quality of the reconstructed images generated by our proposed scheme with the modified quantization table is quite close to others because our hiding strategy does not cause significant modifications on DCT coefficients. To demonstrate the relationship between the hiding capacity and image quality of the reconstructed images by using our proposed data hiding strategy with different sub-areas in the middle frequency of the

(a) Boats (32.94dB) (b) Jet (34.57dB) (c) Lena (35.35dB)

(d) Mandrill (27.77dB) (e) Peppers (35.22dB) (f) Zelda (35.94dB)

FIGURE 14. Image quality of reconstructed images with standard quantization table

DCT-based image, the experimental results obtained by using our proposed scheme with the modified quantization table is shown in Table 4.

TABLE 4. Comparison of mean PSNRs of the six stego-images with modified and different sub-areas of coefficients in the DCT-based images

Cover images	Our scheme (+ABW sub-areas)		Our scheme (+CDE sub-areas)		Our scheme (+ABCDE sub-areas)		Our scheme (+ABWCDE sub-areas)	
	Hiding capacity	PSNR	Hiding capacity	PSNR	Hiding capacity	PSNR	Hiding capacity	PSNR
Boats	36864	33.13	53248	33.10	65536	33.06	90112	33.05
F16	36864	34.66	53248	34.62	65536	34.56	90112	34.53
Lena	36864	35.44	53248	35.39	65536	35.32	90112	35.28
Mandrill	36864	28.25	53248	28.24	65536	28.22	90112	28.22
Peppers	36864	35.25	53248	35.20	65536	35.13	90112	35.09
Zelda	36864	35.86	53248	35.80	65536	35.74	90112	35.69

From Table 4, we can see that the most significant difference between the best and worst cases for a given reconstructed image in its image quality is about 0.08 dB. However, when the number of sub-areas in the middle frequency in the DCT-based image increases, the hiding capacity increases significantly. Take the image "Boats" for example. The hiding

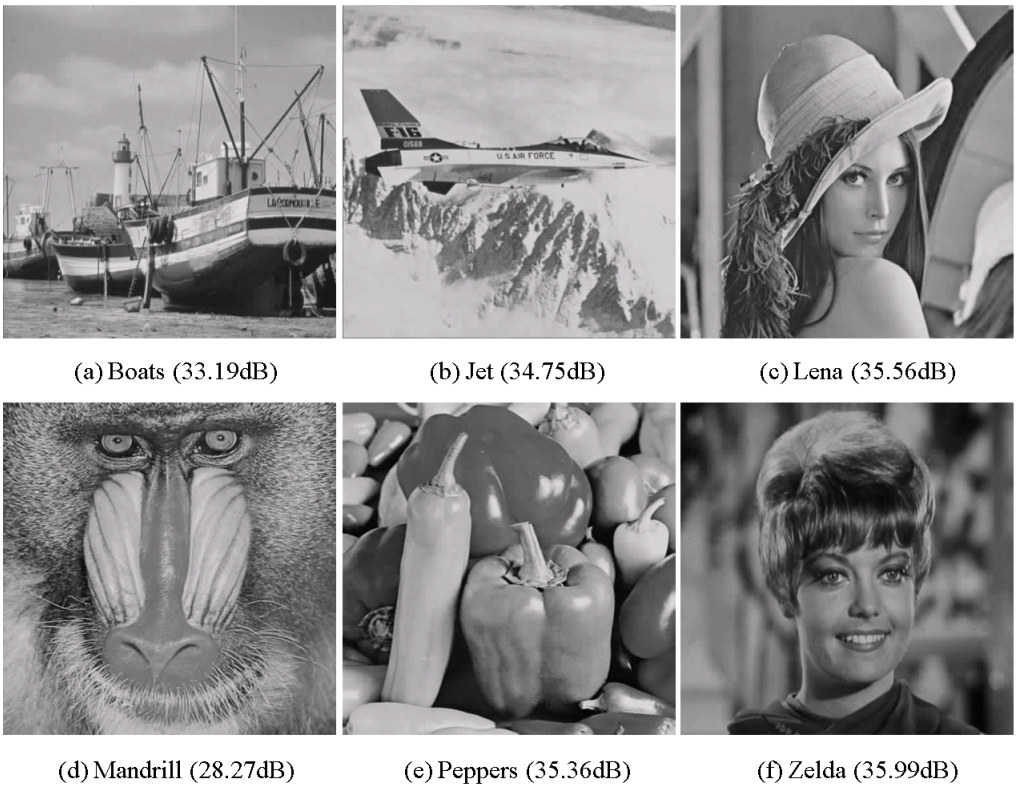


FIGURE 15. Image quality of reconstructed images with the modified quantization table

capacity is 36864 bits when the three A, B and W sub-areas are used for data hiding. When the six A, B, W, C, D and E sub-areas are used for data hiding, the hiding capacity can be as much as 90112 bits while still maintaining image quality at 33.05 dB.

To further prove our scheme's performance on hiding capacity, we compared our proposed scheme with three existing schemes: Chang et al.'s [2], Lin and Shiu's [14] and Lin et al. [15] schemes. Note that the hiding capacity of Iwata et al.'s scheme [12], which is based on the experimental results demonstrated in Chang et al.'s paper, the largest hiding capacity that can be achieved by Iwata et al. is 40000 bits. Table 5 makes it clear that our proposed scheme not only outperforms the other three existing schemes but also outperforms Iwata et al.'s scheme in hiding capacity, which is 40000 bits. Note the experimental results regarding Lin et al.'s scheme [15] shown in Tables 5 and 6 are referred to their original experimental results in their paper.

To prove our proposed scheme successfully provides higher hiding capacity than other existing schemes without at the cost of less image quality of the stego-images, the corresponding *PSNR*s offered by our proposed scheme and three existing schemes are shown in Table 6. From Table 6, we can see the image quality of our proposed scheme with the modified quantization table is higher than that offered by other existing schemes.

5. **Conclusions.** DCT is a widely used mechanism for frequency transformation. Although Lin and Shiu designed a reversible data hiding scheme that offers about 60000 bits of hiding capacity and achieves reversibility, the reversibility function may limit the hiding capacity of DCT-based images. If reversibility is not required, a DCT-based image might offer larger hiding capacity and maintaining image quality higher than that offered

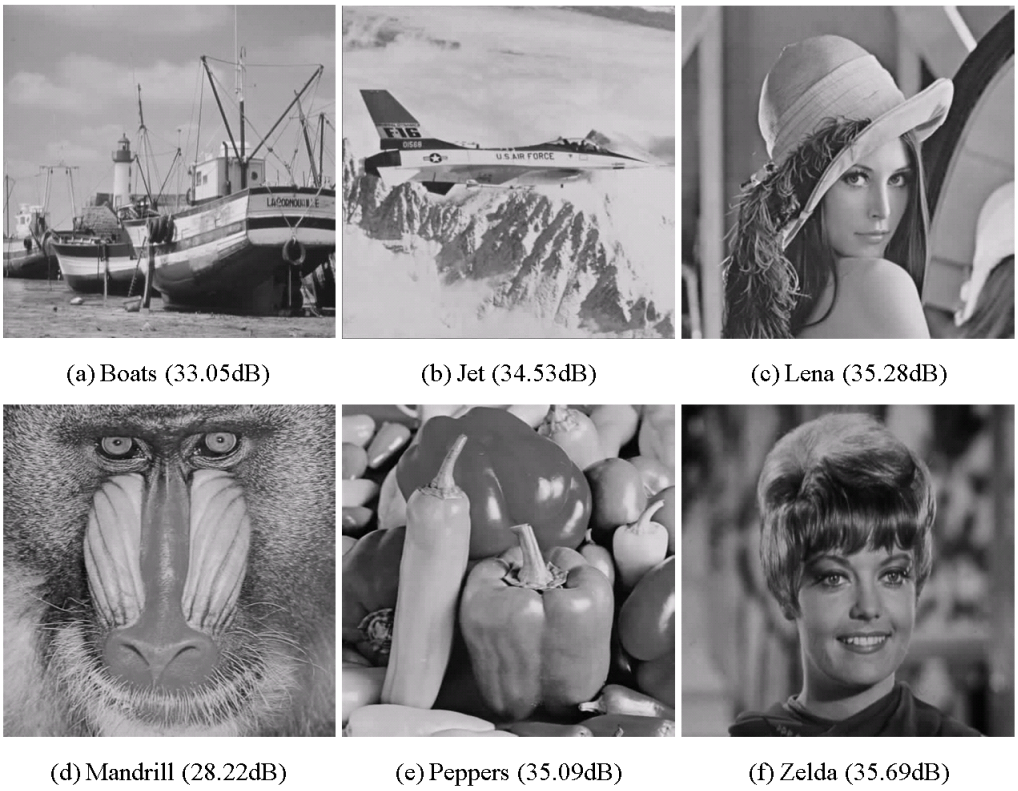


FIGURE 16. Image quality of reconstructed images with our proposed scheme and the modified quantization table when six sub-areas are used for data hiding

TABLE 5. Comparison of hiding capacity of the six stego images among our schemes and three existing schemes

Cover images	Chang et al.'s scheme [2] L=9		Lin et al.'s scheme [15] D=4	Lin and Shiu's scheme [14]		Our scheme (+ ABWCDE sub-areas)	
	Standard quantization table	Modified quantization table	Modified quantization table	Standard quantization table	Modified quantization table	Standard quantization table	Modified quantization table
Boats	36817	36710	–	58357	58426	90112	90112
F16	36852	36817	–	57889	57966	90112	90112
Lena	36861	36850	38834	57644	57717	90112	90112
Mandrill	36094	35402	30734	66048	66075	90112	90112
Peppers	36842	36804	–	56936	56981	90112	90112
Zelda	36864	36861	–	55184	55190	90112	90112

by Lin and Shiu's and other existing DCT-based schemes. To explore these two issues, we propose a notation system-based data hiding scheme in this paper.

By dividing the coefficients in the six sub-areas in the middle frequency in the DCT-based image, our proposed scheme allows senders to adaptively embed secret data into coefficients. Even if only five sub-areas (A, B, C, D, E) are used for data hiding, the

TABLE 6. Comparison of PSNRs of the six stego images among our schemes and three existing schemes

Cover images	Chang et al.'s scheme [2] L=9		Lin et al.'s scheme [15] D=4	Lin and Shiu's scheme [14]		Our scheme (+ ABWCDE sub-areas)	
	Standard quantization table	Modified quantization table	Modified quantization table	Standard quantization table	Modified quantization table	Standard quantization table	Modified quantization table
Boats	27.49	29.75	–	31.95	32.26	32.81	33.05
F16	27.73	29.98	–	33.21	33.44	34.36	34.53
Lena	28.13	30.34	26.77	34.01	34.30	35.09	35.28
Mandrill	24.22	26.46	23.13	26.75	27.27	27.73	28.22
Peppers	28.54	30.65	–	34.10	34.30	34.96	35.09
Zelda	29.5	31.64	–	35.28	35.36	35.64	35.69

hiding capacity of our scheme is almost always higher than with other existing schemes under typical circumstances. Experimental results also confirm that even when six sub-areas are used for data hiding, the image quality of the stego-image is still higher than that offered by other existing schemes. Moreover, our scheme has the same advantage as other existing DCT-based schemes; that is the security of the hidden data is guaranteed because the secret data are embedded into the DCT coefficients rather than pixels of a cover image. Based on the results of our proposed scheme, our next step is to add the reversibility function back into the data hiding scheme to expand the applications of our proposed scheme.

Acknowledgment. Some of the results demonstrated in this paper are sponsored by the National Science Council, Project Number NSC-97-2221-E-126-010.

REFERENCES

[1] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, Lossless Generalized-LSB Data Embedding, *IEEE Trans. Image Processing*, vol. 14, no. 2, pp. 253-266, Feb. 2005.

[2] C. C. Chang, C. C. Lin, C. S. Tseng, and W. L. Tai, Reversible hiding in DCT-based compressed images, *Information Sciences*, vol. 177, pp. 2768-2786, July 2007.

[3] C. C. Chang, T. S. Chen, and L. Z. Chung, A Steganographic Method Based upon JPEG and Quantization Table Modification, *Information Sciences*, vol. 141, 2002, pp. 123-138.

[4] C. C. Chang, J. Y. Hsiao, and C. S. Chan, Finding Optimal Least-Significant-Bit Substitution in Image Hiding by Dynamic Programming Strategy, *Pattern Recognition*, vol. 36, no. 7, pp. 1595-1683, July 2003.

[5] C. C. Chang, W. L. Tai, and C. C. Lin, A Reversible Data Hiding Scheme Based on Side Match Vector Quantization, *IEEE Trans. Circuits and Systems for Video Technology*, vol. 16, no. 10, pp. 1301–1308, 2006.

[6] C. C. Chang, Y. H. Chen, and C. C. Lin, A Data Embedding Scheme for Color Images Based on Genetic Algorithm and Absolute Moment Block Truncation Coding, *Soft Computing*, vol. 13, pp. 21–331, October 2008.

[7] K. L. Chung, C. H. Shen and L. C. Chang, A Novel SVD- and VQ-based Image Hiding Scheme, *Pattern Recognition Letters*, vol. 22, no. 9, pp. 1051–1058, 2001.

[8] B. V. Dasarathy, Image Data Compression: Block Truncation Coding, *IEEE Computer Society Press*, pp. 164–173, 1995.

[9] W. Diffie and M. E. Hellman, Exhaustive Cryptanalysis of the NBS Data Encryption Standard, *IEEE Computers*, vol. 10, pp. 74–84, 1977.

[10] J. Fridrich, M. Goljanb, R. Du, Invertible Authentication Watermark for JPEG Images, *IEEE Int. Conf. Information Technology: Coding and Computing*, pp. 223–227, 2001.
[11] M. Ghanbari, Standard Codecs: Image Compression to Advanced Video Coding, IET, 2003.
[12] M. Iwata, K Miyake, and A. Shiozaki, Digital Steganography Utilizing Features of JPEG Images, *IEICE Trans. Fundamentals*, vol. E87-A, no. 4, pp. 929–936, April 2004.
[13] H. Kobayashi, Y. Noguchi and H. Kiya, A Method of Embedding Binary Data into JPEG Bitstreams, *IEICE Trans. Fundamentals*, vol. 6, pp. 1469–1476, 2000.
[14] C. C. Lin and P. F. Shiu, DCT-based reversible data hiding scheme, *Proc. of the 3rd International Conference on Ubiquitous Information Management and Communication (ICUIMC'09)*, pp. 327–335, 2009.
[15] C. Y. Lin ,C. C. Chang, and Y. Z. Wang, Reversible Steganographic Method with High Payload for JPEG Images, *IEICE Trans. Information and Systems*, vol. 91-D , no. 3, pp. 836–845, 2008.
[16] Y.K. Lee and L. H. Chen, High Capacity Image Steganographic Model, *Proc. of IEEE International Conference on Vision, Image and Signal Processing*, vol. 147, no. 3, pp. 288–294, 2000.
[17] Z. Ni, Y.Q. Shi, N. Ansari, W. Su, Reversible Data Hiding, *Proc. of International Symposium on Circuits and Systems (ISCAS '03)*, vol. 2, pp. 912–915, May 2003.
[18] R.L. Rivest, A. Shamir, L. Adelman, A Method for Obtaining Digital Signatures and Public-key Cryptosystem, *Communications of the ACM*, vol. 21, no. 2, pp. 120-126, Feb. 1978.
[19] J. Tian, Reversible Data Embedding Using a Difference Expansion, *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890-896, Aug. 2003.
[20] P. Tsai, Y. C. Hu and C. C. Chang, An Image Hiding Technique Using Block Truncation Coding, *Proc. of Pacific Rim Workshop on Digital Steganography*, pp. 54-64, July 2002.
[21] R. Z. Wang, C. F. Lin and J. C. Lin, Image Hiding by Optimal LSB Substitution and Genetic Algorithm, *Pattern Recognition*, vol. 34, no. 3, pp. 671–683, 2001.
[22] G. Xuan, J. Zhu, J. Chen, Y.-Q. Shi, Z. Ni, and W. Su, Distortionless Data Hiding Based on Integer Wavelet Transform, *IEE Electronics Letters*, vol. 38, no. 25, pp. 1646-1648, Dec. 2002.