High-Quality and High-Capacity Data Hiding Based on Absolute Moment Block Truncation Coding

Yung-Yao Chen[1], Chih-Hsien Hsia[2], Kuan-Yu Chi[1], Bo-Yan Chen[1]

[1] Grad. Inst. of Automation Technology, National Taipei Tech. University, Taiwan

[2] Dept. Computer Science and Information Eng., National Ilan University, Taiwan

yungyaochen@mail.ntut.edu.tw, chhsia625@gmail.com, t104618011@ntut.edu.tw, t105618004@ntut.edu.tw

Abstract

Numerous images are transmitted over the Internet every day. However, for some sensitive images, such as personal medical images, public networks constitute an insecure environment. Watermarking methods are therefore essential for ensuring information security. Additionally, most digital transmitted images are compressed first so as to increase transmission efficiency. This paper presents a data hiding method for absolute moment block truncation coding (AMBTC)-compressed images. The goal of this method is to improve the Internet transmission efficiency by compressing the image and to ensure secure communication by embedding extra secret information into the compressed image at the same time. In the proposed method, the cover image is first classified into two types of blocks through a block classification scheme. Several embedding rules are proposed according to the block type, and an iteration-based halftoning method is integrated with AMBTC to optimize the image quality. Moreover, the blind decoding method is proposed wherein the secret data are extracted from the stego image itself. Experimental results showed that this method outperformed other state-of-the-art methods in terms of image quality and embedding capacity.

Keywords: Multimedia watermarking, Information security, Absolute moment block truncation coding (AMBTC)

1 Introduction

Watermarking, also referred to as data hiding, has gained increasing attention from researchers over the past few decades [1] and has become an important research topic with the proliferation of images transmitted over the public Internet. In watermarking techniques, secret data are embedded into the cover images. The secret data can only be extracted by legitimate users who possess a secret key, such as having knowledge of the embedding rules. Watermarking provides a solution for image authentication [2], in which the secret data can be the authentication code that is used to verify the authenticity of the received image.

Most watermarking schemes can be roughly divided into three types: (1) frequency-domain methods [3-4], (2) spatial-domain methods [5-8], and (3) compression-domain methods [9]. Images that are embedded with secret data are usually called the *cover images*, and the images that embed secret data are usually called the *stego images*.

Regarding the frequency-domain schemes, the original image is first transformed into frequency coefficients by using transformation methods such as discrete wavelet transformation [3] and discrete cosine transformation [4]. Next, the secret data are embedded by adjusting the frequency components. Regarding the spatial-domain schemes, the secret data are embedded by adjusting the pixel value strategically by considering image details and operations such as the least significant bit (LSB) [5], pixel value difference or expansion [6], prediction error [7], and histogram shifting [8]. For the compression-domain methods, the watermarking schemes basically depend on how the original images are compressed. For images compressed using a frequency-domain approach, watermarking is achieved by using a concept similar to that of the frequency-domain methods [9]. By contrast, for images compressed using a spatial-domain approach, such as absolute moment block truncation coding (AMBTC), watermarking is achieved based on a concept similar to that of the spatial-domain methods. Because it is most related to the present study, the literature review for AMBTC-based watermarking is discussed in Section 2.

Image quality is another important concern related to watermarking. However, most data hiding schemes result in inevitable distortion during the hiding operation; that is, the image quality of a stego image is poorer than that of the original cover image because of the embedded secret data. Thus, watermarking schemes with reversible properties have been proposed [10-11]. The loss of image quality is even worse in compression-domain watermarking because the image

[*] Corresponding Author: Yung-Yao Chen; E-mail: yungyaochen@mail.ntut.edu.tw

DOI: 10.3966/160792642019032002007

quality is degraded twice, once because of the compression procedure followed by the watermarking procedure. Reversible compression-domain watermarking schemes have thus been proposed [12-13]. However, their embedding capacity is usually lower than that of other watermarking schemes. Moreover, the quality of compressed image is limited due to compression process.

In this work, we present a novel AMBTC-based watermarking scheme that seeks high-quality stego images. Watermarking in compressed images has its unique advantage: Most digital images on the Internet are compressed for ensuring high transmission efficiency. For steganography, concealing secret data in an undetectable manner is essential. If stego images are uncompressed, they might be vulnerable to invaders. Concurring with the concern of image quality, maintaining high stego image quality also indicates imperceptivity.

Several halftoning-based BTC methods have recently been proposed, such as ordered dither BTC [14] and direct binary search BTC (DBSBTC) [15]. Digital halftoning is a technique that converts a grayscale image into a two-level binary image. Because the human visual system (HVS) model is involved, halftoning-based BTC methods are effective in alleviating the possibly troublesome blocking effect. In this study, watermarking was integrated with halftoning-based BTC.

The rest of this paper is organized as follows. In Section 2, the related works are reviewed. Section 3 presents the proposed AMBTC-based watermarking method and its corresponding data extraction scheme. The experimental results are discussed and compared with those of the state-of-the-art methods in Section 4. The conclusions are presented in Section 5.

2 Related Works

2.1 Absolute Moment Block Truncation Coding (AMBTC) [16]

AMBTC [16] is an efficient lossy image compression method, which is popular because of its low computational cost. The input grayscale image is divided into nonoverlapped $n \times n$ blocks. Each block is compressed using one bit plane and two quantization levels. The block size (n) can be freely selected by the users. In this study, for simplicity and without lack of generality, the default block size was set to $n = 4$. Note that the proposed method is compatible to other block size. In each block, the mean grayscale value ($\bar{x}$) can be calculated by

$$\bar{x} = \frac{1}{16}\sum_{i=1}^{16} x_i \,, \tag{1}$$

where x_i denotes the i-th pixel value within this block, $i = 1, 2, ..., 16$. The bit plane (B_i) is obtained by comparing each pixel value of this block with $\bar{x}$. For the case that the pixel value is less than $\bar{x}$, the bit plane value is recorded as “0”; otherwise, it is recorded as “1.” For AMBTC quantization, the two quantization levels, the high mean (H_i) and low mean (L_i) values, can be calculated by

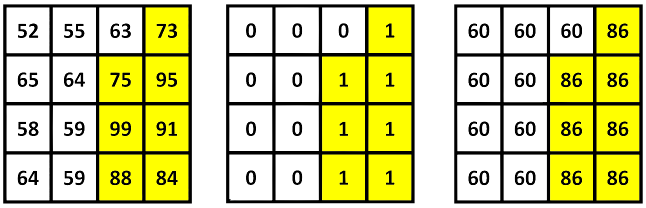
$$H_i = \frac{1}{16-q}\sum_{x_i \geq \bar{x}} x_i \tag{2}$$

and

$$L_i = \frac{1}{q}\sum_{x_i < \bar{x}} x_i \,, \tag{3}$$

where q indicates the number of pixels whose pixel value is less than $\bar{x}$. The AMBTC method maintains the first absolute moment of each original grayscale image block.

Figure 1 illustrates an example of how the AMBTC method works. For each block, its compression code of comprises a trio (H_i, L_i, B_i), and 32 bits are required to store this trio: two quantization levels (8 bits + 8 bits) and one bit plane (16 bits). Compared with the storage required for saving the original image block that requires 16 × 8 = 128 bits, AMBTC requires only one-fourth of the original storage. Receivers can decode by recovering the bit plane with H_i and L_i.



52	55	63	73
65	64	75	95
58	59	99	91
64	59	88	84

(a) Original block

0	0	0	1
0	0	1	1
0	0	1	1
0	0	1	1

(b) Bit plane with two corresponding levels $H_i = 86, L_i = 60$

60	60	60	86
60	60	86	86
60	60	86	86
60	60	86	86

(c) Reconstructed block

Figure 1. Illustration of AMBTC scheme

2.2 Recent AMBTC-Based Watermarking Methods

Since AMBTC is a common compression method with low computation loads, some AMBTC-based watermarking methods are recently proposed to enhance information security [12-13, 17-25]. Ou and Sun proposed an AMBTC-based data hiding method with the property of blind decoding [17]. First, a threshold *Thr* is predefined for block classification. Each block of the initial AMBTC compressed image is classified into two types as follows:

$$block\ type = \begin{cases} smooth,\ if\ \left|H_i - L_i\right| \le Thr \\ complex,\ otherwise \end{cases}. \tag{4}$$

Each of the complex blocks is embedded into 1-bit secret data by simultaneously reversing the bit plane and exchanging the order of H_i and L_i in the compression trio code. That is, if the secret code is 0, the original compression trio code (H_i , L_i , B_i) is maintained. By contrast, if the secret code is 1, the trio code becomes (L_i , H_i , $\overline{B}_i$), where $\overline{B}_i$ denotes the reversed bit plane. Each of the smooth blocks is embedded into 16-bit secret data by completely replacing the bit plane. The two quantization levels are updated according to the embedded secret data. Hong *et al.* proposed an AMBTC-based watermarking method that also involves a block classification step [18]. Moreover, Hu et al. combined watermarking and AMBTC for image authentication [19]. Chang et al. proposed an AMBTC-based watermarking method, where the histogram is modified to embed the secret message [20]. Chang et al. used a dynamic programming strategy for AMBTC-based watermarking [21]. Furthermore, Yang and Yin combined a chaotic map with AMBTC for removable visible watermarking [22]. Liu et al. proposed a minima–maxima-preserving approach for maintaining the quality of the de derived AMBTC stego image.

2.3 Our Contribution

Considering the aforementioned drawbacks, we propose a novel AMBTC-based watermarking method.

- We propose a watermarking method that jointly integrates the hiding operation with DBSBTC. During the process of embedding the secret data, the AMBTC bit plane of each block is constantly modified according to the local image content of the cover image and HVS.
- We modified DBSBTC to ensure that the embedded secret data are not damaged during compression. DBSBTC substantially improves the image quality in traditional AMBTC.
- We designed two embedding schemes, one of which is used in an AMBTC block, depending on the image content and block type. Therefore, large secret data can be embedded while ensuring minimal image distortion.
- Unlike [12], this study provides a blind decoding method; that is, secret data are extracted without requiring additional information. This method expands the flexibility for real security-related applications.

3 Proposed Method

Figure 2 illustrates the flow diagram of the proposed AMBTC-based data hiding method. In [17], Ou and Sun utilized the difference between two quantization levels $\left|H_i - L_i\right|$ to distinguish between smooth and complex blocks; however, this difference has no correlation with the hiding operation. The proposed method completely utilizes the quantization-level difference, not only for block classification but also for embedding the secret data.

The input cover image is divided into 4 × 4 nonoverlapped blocks. For each block, the initial quantization levels are calculated by (1)-(3), and the blocks are divided into two categories according to (4). To emphasize the effect of the threshold in (4), it is hereby referred to as *just noticeable intensity shift* (thr_{JNIS}), which indicates the blocks where the quantization difference cannot be perceived. The default value of thr_{JNIS} is set to 4.

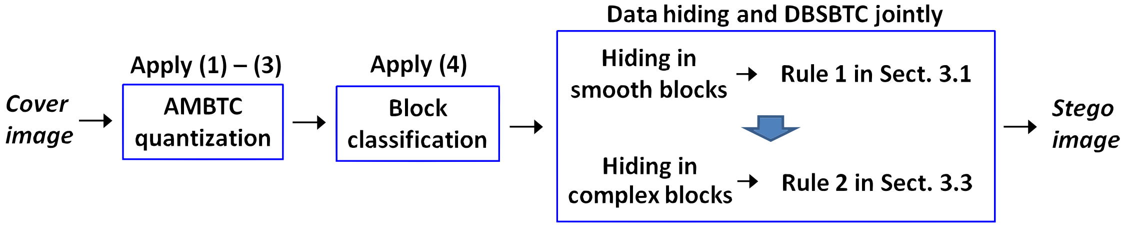


Figure 2. The flow diagram of the proposed scheme: the data hiding phase

3.1 Rule 1: Watermarking in Smooth Blocks

After the block classification step, the blocks are divided into smooth blocks and complex blocks. In [17], because the bit plane of a smooth block was totally replaced by 16-bit secret data, the perceptual difference within thr_{JNIS} can be compromised for embedding the secret data. Therefore, we extend [17] by further modifying the values of H_i and L_i ; however, the difference between them is still within thr_{JNIS} . To embed more data in a smooth block, we introduce a term T_1 , which is less than or equal to thr_{JNIS} and is a power of two. For example, as $thr_{JNIS} = 4$, T_1 can be 4, 2, and 1.

The hiding capacity of a smooth block is $16 + \log_2 T_1 + 1$ bits, as shown in Figure 3. For each smooth block, 16 secret bits are embedded by replacing the bit plane, and $\log_2 thr_1$ secret bits are embedded by modifying the initial H_i and L_i. When $T_1 = thr_{JNIS} = 4$, additional 2-bit data are embedded as follows:

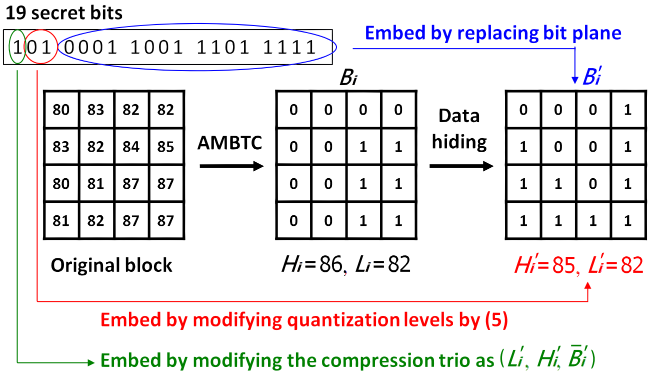


Figure 3. Example of data hiding in a smooth block

$$\begin{cases} code\ 00,\ H_i' = m + \dfrac{thr_{JNIS}}{2}, L_i' = m - \dfrac{thr_{JNIS}}{2} \\ code\ 01,\ H_i' = m + \dfrac{thr_{JNIS}}{2} - 1, L_i' = m - \dfrac{thr_{JNIS}}{2} \\ code\ 10,\ H_i' = m + \dfrac{thr_{JNIS}}{2} - 1, L_i' = m - \dfrac{thr_{JNIS}}{2} + 1 \\ code\ 11,\ H_i' = m + \dfrac{thr_{JNIS}}{2} - 2, L_i' = m - \dfrac{thr_{JNIS}}{2} + 1 \end{cases}, \quad \textbf{(5)}$$

where $m = \left\lceil \dfrac{H_i + L_i}{2} \right\rceil$ is the rounded-up nearest integer and the pair (H_i', L_i') indicates the modified quantization levels. The difference between the modified quantization levels ranges from 1 to thr_{JNIS}; therefore, the additional 1-bit data are embedded by possibly reversing the bit plane and switching the order of the two quantization levels simultaneously. If the secret code is 0, the original compression trio code (H_i, L_i, B_i) is maintained. By contrast, if the secret code is 1, the trio code becomes (L_i, H_i, $\bar{B}_i$).

3.2 Concept of DBSBTC Method

This subsection briefly presents the concept of the DBSBTC algorithm (for more details, please refer to [15] and [21]). In the next subsection, watermarking and DBSBTC are performed jointly in the complex blocks.

Figure 4 shows the conceptual flow diagram of DBSBTC comprising two steps, bit plane modification and quantization level modification. Given the original image block $g[m,n]$ and a randomly initialized bit plane $B[m,n]$, as well as (H_i, L_i) from AMBTC quantization, DBSBTC performs a pixel-based manipulation that constantly modifies $B[m,n]$ until minimal image distortion is achieved. The HVS model is involved in DBSBTC; thus, the perceived error image is defined as

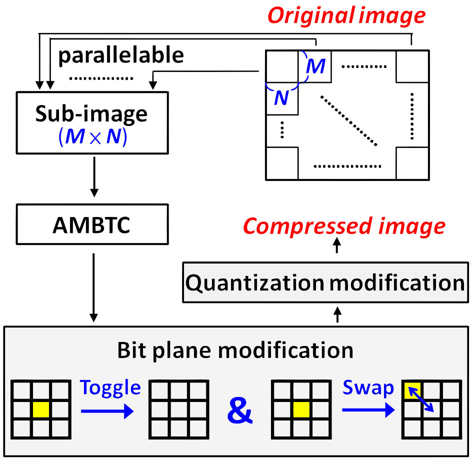


Figure 4. Conceptual flow diagram of the DBSBTC method

$$\tilde{e}(x,y) = \sum_{m,n} e[m,n] P_{HVS}(x - mX, y - nY), \quad \textbf{(6)}$$

where P_{HVS} indicates the spatial-domain HVS model, $e[m,n]$ indicates the error image, and (X,Y) indicates the basis for the lattice of addressable dots. To estimate the image distortion, the HVS-based error metric is defined by

$$\phi = \int_x \int_y \left|\tilde{e}(x,y)\right|^2 dxdy\,. \quad \textbf{(7)}$$

To modify the bit plane until the minimal value of (7) is achieved, DBSBC iteratively performs toggle and swap operations at each current processing pixel. If a toggle operation is performed, the pixel value of the temporary B[m, n] is changed to its opposite, implying that for the compressed image, the high and low means are exchanged at that pixel. If a swap operation is performed, the pixel value (of the current pixel position) is switched with the value of its eight neighbors. Among all possible changes, only the updated bit plane corresponding to the maximum reduction of ϕ is accepted.

Trial changes are iteratively tested on a pixel-by-pixel basis in a raster order throughout this block until minimal error is found. Minimal image distortion is achieved as the minimal error of each block is achieved. Although (6) and (7) are defined in the spatial domain, before the compressed image is produced, the high and low mean values are rounded off to the closest integers according to the AMBTC format.

3.3 Rule 2: Watermarking in Complex Blocks

After data are embedded into the smooth blocks, watermarking and DBSBTC are performed jointly in the complex blocks. To embed data, the DBSBTC algorithm is modified.

The hiding capacity of a complex block is $2 + \log_2 T_2 + 1$ bits. For using Rule 2, 2-bit secret data are embedded by applying the LSB approach at two quantization levels separately. To embed data during bit plane modification, Rule 2 introduces the term weighted sum (W). As illustrated in Figure 5, the bit plane of each 4 × 4 block contains 16 pixel values $b_i \in \{0,1\}$, $i = 1,...,16$. The weight map is defined by assigning each pixel position a unique weight $w_i = i$, as shown in Figure 5(b).

Alternatively, the order of w_i can be permutated using the secret key K to improve information security. The weighted sum (W) is calculated by

$$W = \sum_{i=1}^{16} b_i \times w_i, \tag{8}$$

As an example, Figure 5(c) shows a bit plane whose weighted sum equals 8.

b_1	b_2	b_3	b_4
b_5	b_6	b_7	b_8
b_9	b_{10}	b_{11}	b_{12}
b_{13}	b_{14}	b_{15}	b_{16}

(a) The order of bit values

w_1	w_2	w_3	w_4
w_5	w_6	w_7	w_8
w_9	w_{10}	w_{11}	w_{12}
w_{13}	w_{14}	w_{15}	w_{16}

(b) Weight map

1	0	1	1
0	0	0	0
0	0	0	0
0	0	0	0

(c) Bit plane whose weighted sum equals eight

Figure 5. Illustration of calculating the weighted sum

In the proposed method, additional $\log_2 T_2$ bits are embedded by applying the modulo operation to W with the divisor of T_2, which is a power of two. When $T_2 = 2$, the 1-bit secret code is embedded by

$$secret\ code = \begin{cases} 0, & if\ W \bmod(2) = 0 \\ 1, & otherwise \end{cases}. \tag{9}$$

Therefore, the secret code can be embedded by modifying the bit plane because each bit plane has its own unique W value.

Once DBSBTC generates the optimal bit plane, the embedding Rule 2 is applied as follows:

Step 1. The 2-bit secret data are embedded by applying the LSB approach at two quantization levels separately.

Step 2. The weighted sum is calculated using (8).

Step 3. Because the secret code is embedded using (9), if the quotient of W divided by 2 is the desired secret code, the current bit plane is left unchanged. Otherwise, the bit plane is modified by performing the toggle or swap operation once. If a toggle operation is performed at the position corresponding to the weight w_i, the weighted sum is changed by adding (or subtracting) w_i. If a swap operation is performed between the positions corresponding to w_i and w_j, W is changed by adding (or subtracting) $w_i - w_j$.

Step 4. To maintain the original bit plane as much as possible, only one additional toggle (or swap) is performed in Rule 2. Consider, for example, Figure 5(c); if the to-be-embedded secret code is 1, the bit plane can be modified by toggling at the positions corresponding to an odd weight (i.e., w_1, w_3, ...) or by swapping between the positions corresponding to an odd weight and an even weight (e.g., w_1 and w_2). Among all trial changes, only the updated bit plane corresponding to the minimum increase in (7) is accepted.

Step 5. Because the quantization levels are different, additional 1-bit data are embedded by possibly reversing the bit plane.

3.4 Summary of the Proposed Algorithm

The stego image construction process is described as below:

Step 1. The original cover image is first divided into 4 × 4 nonoverlapped blocks.

Step 2. According to (4), each block is classified as either a smooth block or a complex block.

Step 3. The secret codes are embedded into each block. As illustrated in Figure 3, DBSBTC has inherent parallel characteristics of block-based processing. To take advantage of this characteristics of block-based processing. To take advantage of this benefit, each block is processed with DBSBTC independently and simultaneously but with different modifications according to the block type.

(i) For a smooth block, neither toggle nor swap is performed. The bit plane and updated quantization levels are determined by the embedding Rule 1.

(ii) For a complex block, typical DBSBTC is performed, followed by the embedding Rules 2 and 3. The result that leads to a lower value of (7) is accepted.

Step 4. The stego AMBTC-compressed image is constructed after each block is processed.

Process of decoding the stego image: Figure 6 illustrates the flow diagram of the decoding phase of the proposed AMBTC-based data hiding method. In the receiver, the secret data are extracted using a blinding decoding method described as below:

Step 1. The received stego image is first divided into 4 × 4 nonoverlapped blocks.

Step 2. All the blocks are arranged in order and each block type is identified through (4).

Step 3. For a smooth block, the $16 + \log_2 T_1 + 1$ hidden data can be extracted by recalling Rule 1 (Section 3.1). For a complex block, the $2 + \log_2 T_2 + 1$ hidden data are extracted by recalling Rule 2 (Section 3.3).

Step 4. Step 3 is repeated until all blocks are processed and the secret data are completely extracted.

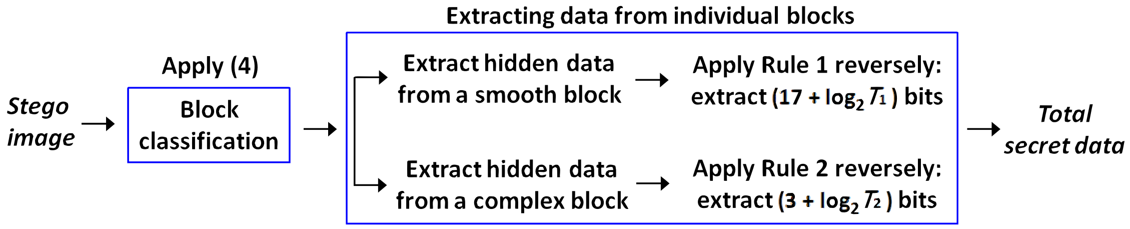
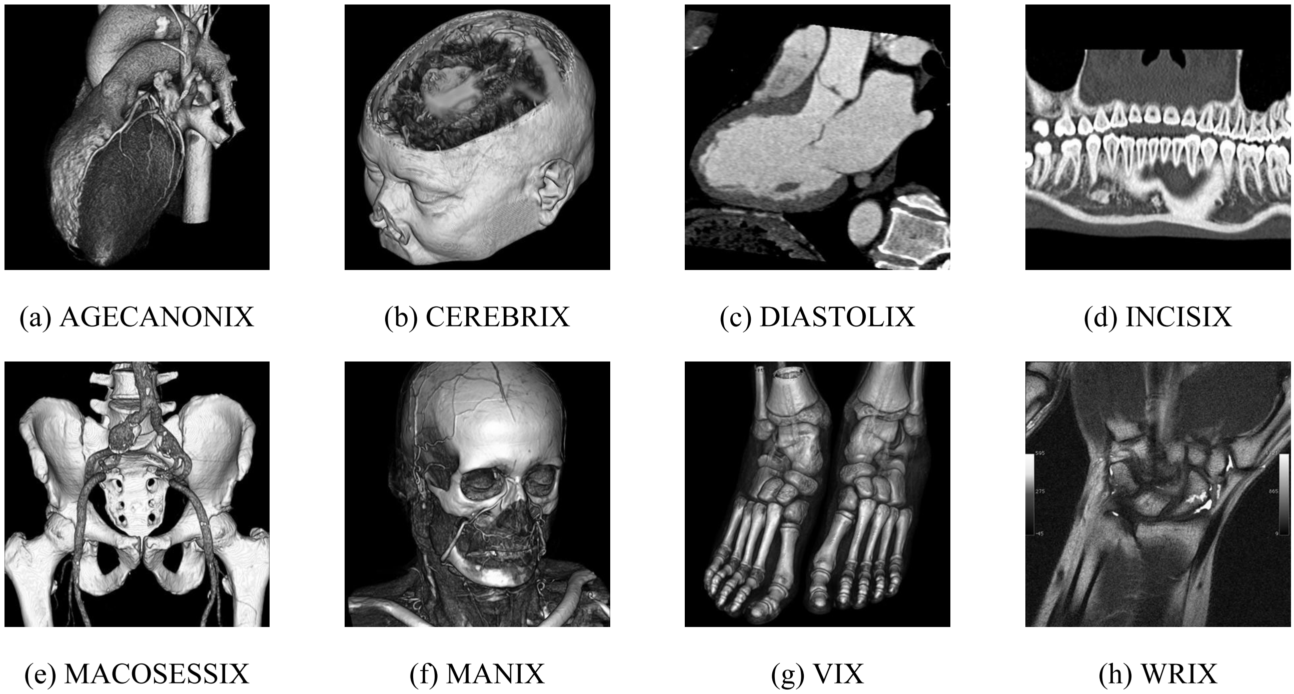

Extracting data from individual blocks
Stego image
Apply (4)
Block classification
Extract hidden data from a smooth block
Apply Rule 1 reversely: extract (17 + log₂ T1) bits
Extract hidden data from a complex block
Apply Rule 2 reversely: extract (3 + log₂ T2) bits
Total secret data

Figure 6. The flow diagram of the proposed scheme: the decoding phase

4 Experimental Results

This section presents the experimental results and evaluation of the proposed method. For comparison, we also performed watermarking by using the three state-of-the-art AMBTC-based methods in [12, 17], and [18]. Because watermarking is usually applied in sensitive personal images, we selected eight medical images (AGECANONIX, CEREBRIX. DIASTOLIX, INCISIX, MACOSESSIX, MANIX, VIX, and WRIX) from the USC-SIPI database [25]. The thumbnails of these test images are shown as depicted in Figure 7. In this work, the secret data are all produced using the same pseudorandom generator.


(a) AGECANONIX (b) CEREBRIX (c) DIASTOLIX (d) INCISIX

(e) MACOSESSIX (f) MANIX (g) VIX (h) WRIX

Figure 7. Test images

4.1 Definition of Image Quality Measures

Two common image quality measures were defined and used to evaluate the performance of the different methods. The first measure was the HVS peak signal-to-noise ratio (HPSNR), which is defined as

$$HPSNR = 10\log\left(\frac{P\times Q\times 255^2}{\sum_{P,Q}\left[\sum_{m,n} q_{m,n}(g_{i+m,j+n} - h_{i+m,j+n})\right]^2}\right), \quad \textbf{(10)}$$

where (P,Q) represents the image size and q represents the HVS coefficients. The variables g and h represent the input cover image and the output stego image, respectively. The second measure was the mean structural similarity index measure (MSSIM), which is defined by

$$MSSIM = \frac{1}{M}\sum_{j=1}^{M} SSIM(g_j, h_j), \quad \textbf{(11)}$$

where the SSIM index is a measure that simulates human visual perception and is used to measure the similarity between two images. In this study, M was set as 3.

4.2 Performance Evaluation and Comparison

Comparing different watermarking methods is difficult because each method has its own parameter settings and different settings result in different performance levels. However, the data capacity typically increases as the image quality decreases. To demonstrate the superiority of the proposed method in detail, we compared this method with other methods separately.

Table 1, Table 2 and Table 3 present the overall comparison results of the proposed method with those proposed in [17] and [18]; these methods were included because they apply the block classification step to classify the compressed blocks into smooth blocks and complex blocks. The propsoed method outperformed those in [17] and [18] in terms of both capacity and image quality (Table 1). The parameter setting of the method in [17] was $thr_{JNIS} = 4$ and that of the method in [18] was $thr_{JNIS} = 4$, with $T_S = 4$, $T_1 = 4$, and $T_2 = 2$. The methods in [17] and [18] as well as the proposed method apply the blind decoding procedure.

Table 1. Overall comparisons of data capacity (in units of bits)

Image	AGECANONIX	CEREBRIX	DIASTOLIX	INCISIX	MACOSESSIX	MANIX	VIX	WRIX
Ref. [17]	80680	59080	62425	65785	59155	63295	72580	12550
Ref. [18]	90104	65624	69415	73087	65709	70401	80924	12890
Ours	236966	199145	203119	208408	200403	204545	218385	120949

Table 2. Overall comparisons of quality measures in terms of HPSNR

Image	AGECANONIX	CEREBRIX	DIASTOLIX	INCISIX	MACOSESSIX	MANIX	VIX	WRIX
Ref. [17]	48.716000	49.589900	50.052200	47.210500	46.135200	49.753600	49.260200	50.149700
Ref. [18]	48.660000	49.496700	49.802900	47.055800	46.044900	49.686900	49.110200	50.133400
Ours	58.851068	59.753800	59.453805	55.653447	54.805668	59.788143	59.162185	58.740142

Table 3. Overall comparisons of quality measures in terms of MSSIM

Image	AGECANONIX	CEREBRIX	DIASTOLIX	INCISIX	MACOSESSIX	MANIX	VIX	WRIX
Ref. [17]	0.989900	0.990800	0.993100	0.993200	0.992000	0.991400	0.994900	0.983100
Ref. [18]	0.989700	0.990700	0.992800	0.993400	0.991800	0.991200	0.995300	0.983000
Ours	0.999989	0.999991	0.999994	0.99999	0.99998	0.999992	0.999992	0.999993

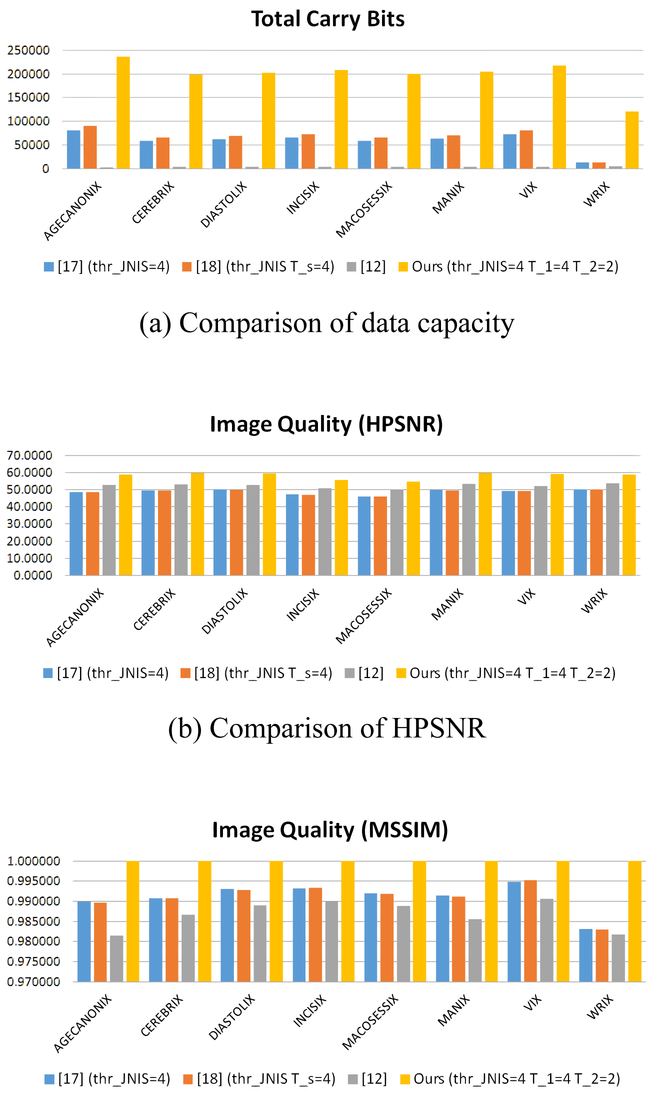
For the method in [17], the difference between the two quantization levels is merely used for block classification. By contrast, the method in [18] applies a quantization level modification technique to embed more data in smooth blocks, and the proposed method applies embedding Rules 1 and 2 to embed more data in the smooth and complex blocks, respectively. Table 1 indicates that the proposed method has the largest data capacity because each type of the blocks is completely ultilized to embed data.

Although the method in [18] has larger data capacity than that in [17], the image quality of the method in [18] is lower than that of the method in [17], as shown in Table 2 and Table 3. This is because when the quantization level modification technique is applied in the smooth blocks, it degrades the image quality because of the embedded random secret data. In the proposed method, although the image quality degrades as the data are embedded into the smooth blocks, the overall image quality is greatly improved because of the following modified DBSBTC procedure. For the proposed embedding Rule 2, while the secret data are embedded into the complex blocks, the modified DBSBTC procedure seeks the optimal bit plane for the minimum HVS-based error metric. Therefore, Table 1 shows that the propsoed method has the best image quality.

Figure 8 shows the comparison of the proposed method with those in [12, 17], and [18]. The method in [12] is slightly different from the other three methods because it does not apply the block classification step. For the method described in [12], the hidden data are embedded by modifying the histogram of each compressed block. The method in [12] is a reversible watermarking approach. However, to extract the original AMBTC image from the stego image, the receiver must have the information of pairs of peak and zeros points; thus, the decoding procedure is not blind. Moreover, the data capacity of the method in [12] is apparently much lower than the other three methods because in [12], the secret data are embedded in the histogram, whereas in the other three methods, the hidden data are embedded in the bit planes.

5 Conclusion

This study presents a novel AMBTC-based watermarking method that combines DBSBTC and watermarking, thereby maintaining high image quality while being embedded with large data. Compared with other state-of-the-art methods, the proposed method demonstrated superiority in terms of data capacity and HPSNR and MSSIM values. Currently, with the proliferation of images transmitted over the Internet,



(a) Comparison of data capacity

(b) Comparison of HPSNR

(c) Comparison of MSSIM

Figure 8. Overall comparisons among various methods by using the eight test images

multimedia watermarking and information security have become critical. The proposed method can provide a solution for image authentication and copyright protection.

In the future, we intend to develop a reversible watermarking scheme based on this framework, because in some cases, preserving the original image is more important than achieving high image quality. Moreover, the proposed framework must be expanded to color image compression.

References

[1] M. Zhao, J. Pan, S. Chen, Entropy-based Audio Watermarking via the Point of View on the Compact Particle Swarm Optimization, *Journal of Internet Technology*, Vol. 16, No. 3, pp. 485-493, May, 2015.

[2] H. Al-Otum, Semi-fragile Watermarking for Grayscale Image Authentication and Tamper Detection Based on an Adjusted Expanded-bit Multiscale Quantization-based Technique, *Journal of Visual Communication and Image Representation*, Vol. 25, No. 5, pp. 1064-1081, July, 2014.

[3] I. Pan, T. Chang, P. Huang, DWT-based Watermarking Using 2DLDA for Color Images, *Journal of Internet Technology*, Vol. 15, No. 7, pp. 1101-1109, Decmber, 2014.

[4] M. Ali, C. Ahn, M. Pant, A Robust Image Watermarking Technique Using SVD and Differential Evolution in DCT Domain, *Optik*, Vol. 125, No. 1, pp. 428-434, January, 2014.

[5] Z. Xia, X. Wang, X. Sun, B. Wang, Steganalysis of Least Significant Bit Matching Using Multi-order Differences, *Security and Communication Networks*, Vol. 7, No. 8, pp. 1283-1291, August, 2014.

[6] I. Dragoi, D. Coltuc, Local-prediction-based Difference Expansion Reversible Watermarking, *IEEE Tran. Image Processing*, Vol. 23, No. 4, pp. 1779-1790, April, 2014.

[7] I. Dragoi, D. Coltuc, On Local Prediction Based Reversible Watermarking, *IEEE Tran. Image Processing*, Vol. 24, No. 4, pp. 1244-1246, April, 2015.

[8] G. Coatrieux, W. Pan, N. Cuppens-Boulahia, F. Cuppens, C. Roux, Reversible Watermarking Based on Invariant Image Classification and Dynamic Histogram Shifting, *IEEE Tran. Information Forensics and Security*, Vol. 8, No. 1, pp. 111-120, January, 2013.

[9] F. Chen, H. He, Y. Huo, Self-embedding Watermarking Scheme Against JPEG Compression with Superior Imperceptibility, *Multimedia Tools and Applications*, Vol. 76, No. 7, pp. 9681-9712, April, 2017.

[10] T. Hsu, A. Yein, W. Hsieh, N. Chen, J. Chiang, T. Su, Reversible Watermarking Algorithm Based on Embedding Pixel Dependence, *Journal of Internet Technology*, Vol. 13, No. 4, pp. 571-580, July, 2012.

[11] T. Pan, S. Weng, Z. Zhou, S. Chu, J. F. Roddick, Reversible Watermarking Based on Position Determination and Pixel Pairs, *Journal of Internet Technology*, Vol. 17, No. 4, pp. 779-787, July, 2016.

[12] I. Chang, Y. Hu, W. Chen, C. Lo, High Capacity Reversible Data Hiding Scheme Based on Residual Histogram Shifting for Block Truncation Coding, *Signal Processing*, Vol. 108, No. 4, pp. 376-388, March, 2015.

[13] Y. Huang, C. Chang, Y. Chen, Hybrid Secret Hiding Schemes Based on Absolute Moment Block Truncation Coding, *Multimedia Tools and Applications*, Vol. 76, No. 5, pp. 6159-6174, March, 2017.

[14] J. Guo, M. Wu, Improved Block Truncation Coding Based on the Void-and-cluster Dithering Approach, *IEEE Transaction Image Processing*, Vol. 18, No. 1, pp. 211-213, January, 2009.

[15] J. Guo, C. Su, Improved Block Truncation Coding Using Extreme Mean Value Scaling and Block-based High Speed Direct Binary Search, *IEEE Signal Processing Letters*, Vol. 18, No. 11, pp. 694-697, September, 2011.

[16] M. Lema, O. Mitchell, Absolute Moment Block Truncation Coding and Its Application to Color Images, *IEEE Transaction* Communications, Vol. 32, No. 10, pp. 1148-1157, October, 1984.

[17] D. Ou, W. Sun, High Payload Image Steganography with

Minimum Distortion Based on Absolute Moment Block Truncation Coding, *Multimedia Tools and Applications*, Vol. 74, No. 5, pp. 9117-9139, November, 2015.

[18] W. Hong, T. Chen, Z. Yin, B. Luo, Y. Ma, Data Hiding in AMBTC Images Using Quantization Level Modification and Perturbation Technique, *Multimedia Tools and Applications*, Vol. 76, No. 3, pp. 3761-3782, February, 2017.

[19] Y. Hu, C. Lo, W. Chen, C. Wen, Joint Image Coding and Image Authentication Based on Absolute Moment Block Truncation Coding, *Journal of Electron Imaging*, Vol. 22, No. 1, pp. 1-11, January, 2013.

[20] C. Chang, H. Wu, T. Chung, Applying Histogram Modification to Embed Secret Message in AMBTC, *Intelligent Information Hiding and Multimedia Signal Processing*, Kitakyushu, Japan, 2014, pp. 489-493.

[21] C. Chang, Y. Liu, S. Nguyen, A Novel Data Hiding Scheme for Block Truncation Coding Compressed Images Using Dynamic Programming Strategy, *SPIE Conference on Graphic and Image Processing*, Beijing, China, 2014, pp. 1-10.

[22] H. Yang, J. Yin, A Secure Removable Visible Watermarking for BTC Compressed Images, *Multimedia Tools and Applications*, Vol. 74, No. 6, pp. 1725-1739, March, 2015.

[23] C. Kim, D. Shin, L. Leng, C. Yang, Lossless Data Hiding for Absolute Moment Block Truncation Coding Using Histogram Modification, *J. Real-Time Image Proc.*, pp. 1-14, October, 2016.

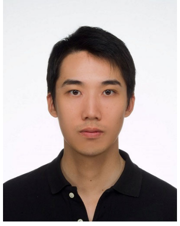
[24] N. Huynh, K. Bharanitharan, C. Chang, Y. Liu, Minima-maxima Preserving Data Hiding Algorithm for Absolute Moment Block Truncation Coding Compressed Images, *Multimedia Tools and Applications*, Vol. 76, No. 3, pp. 1-17, February, 2017.

[25] Y. Chen, K. Chi, Cloud Image Watermarking: High Quality Data Hiding and Blind Decoding Scheme Based on Block Truncation Coding, *Multimedia Systems*, pp. 1-13, July, 2017.

[26] D. Lieberman, Jan P. Allebach, A Dual Interpretation for Direct Binary Search and Its Implications for Tone Reproduction and Texture Quality, *IEEE Transaction Image Processing*, Vol. 9, No. 11, pp. 1950-1963, November, 2000.

[27] *The USC-SIPI Image Database*, http://sipi. usc.edu/database/.

Biographies



Yung-Yao Chen received his Ph.D. (2013) degree in Electrical Engineering from Purdue University, USA. He is currently an Assistant Professor in the Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, Taiwan. His research interests include halftoning, and human-computer interaction.

Chih-Hsien Hsia received the Ph.D. degree in Electrical Engineering from Tamkang University, New Taipei, Taiwan, in 2010. He currently is an Associate Professor with the Department of Computer Science and Information Engineering, National Ilan University, Taiwan. His research interests include DSP IC Design, Multimedia Signal Processing, and Information Education.

Kuan-Yu Chi received his M.S. degree in the Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, Taiwan (2017). His research interests include vcomputer ision, block truncation coding, and digital halftoning.

Bo-Yan Chen received his B.S. degree from Feng Chia University, Taiwan. He is pursuing his M.S. degree in the Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, Taiwan. His research interests include image security, block truncation coding, and digital halftoning.