Copyright ⓒ 2014 KSII

A Semi-fragile Image Watermarking Scheme Exploiting BTC Quantization Data

Dongning Zhao, Weixin Xie
ATR National Defense Technology Key Laboratory, College of Information Engineering
Shenzhen University, Shenzhen, 518060, China
[e-mail: zhaodongning1979@gmail.com]
*Corresponding author: Dongning Zhao

Received December 31, 2013; revised March 14, 2014; accepted March 23, 2014; published April 29, 2014

Abstract

This paper proposes a novel blind image watermarking scheme exploiting Block Truncation Coding (BTC). Most of existing BTC-based watermarking or data hiding methods embed information in BTC compressed images by modifying the BTC encoding stage or BTC-compressed data, resulting in watermarked images with bad quality. Other than existing BTC-based watermarking schemes, our scheme does not really perform the BTC compression on images during the embedding process but uses the parity of BTC quantization data to guide the watermark embedding and extraction processes. In our scheme, we use a binary image as the original watermark. During the embedding process, the original cover image is first partitioned into non-overlapping 4×4 blocks. Then, BTC is performed on each block to obtain its BTC quantized high mean and low mean. According to the parity of high mean and the parity of low mean, two watermark bits are embedded in each block by modifying the pixel values in the block to make sure that the parity of high mean and the parity of low mean in the modified block are equal to the two watermark bits. During the extraction process, BTC is first performed on each block to obtain its high mean and low mean. By checking the parity of high mean and the parity of low mean, we can extract the two watermark bits in each block. The experimental results show that the proposed watermarking method is fragile to most image processing operations and various kinds of attacks while preserving the invisibility very well, thus the proposed scheme can be used for image authentication.

Keywords: Fragile image watermarking, information hiding, blind watermarking, block truncation coding, content authentication

This research was supported by the National Natural Science Foundation of China under grant 61171150 and the Special Fund on Strategic New Industry Development of Shenzhen under grant JCYJ20120817163934173, ZDSY20120613125016389, JCYJ20130329105534856.

http://dx.doi.org/10.3837/tiis.2014.04.019

1. Introduction

Over the last two decades, information hiding has become an emerging technology that embeds secret information in image [1-3], video [4-7] or audio [8,9]. In the digital information network age, the possible applications of information hiding technologies become broader and broader, and its important branches is the digital watermark. The first application coming to mind is copyright protection of digital media. In the digital world, it is possible for almost anyone to duplicate or manipulate digital work without degrading the quality. The embedded watermark permits identification of the owner of the work. The second application is maybe in the field of data security, where watermarks can be used for certification, authentication, and conditional access. For example, certification is an important issue for official documents, and we can hide the identity number that is written in clear text on the card as a digital watermark in the identity photo; therefore switching or manipulating the identity photo will be detected. The third application is the authentication of image content. The goal of this application is to detect any alterations and modifications in an image. The paper focuses on this application.

According to the robustness together with the application of the watermarks, digital watermarking methods can be categorized into three classes: robust, fragile and multipurpose approaches. Robust watermarking approaches [1,2,5-7,10-12] are mainly designed for copyright protection where the watermark is still detectable after accidental or malicious attacks, while fragile watermarking schemes [3,8,13,14] are basically designed for integrity verification and content authentication where the slightest alteration of the image is detectable in the extracted watermark. Recently, several multipurpose watermarking methods [4,15,16] have been proposed to simultaneously achieve multiple purposes such as copyright protection, content authentication, image retrieval and data hiding. And the semi-fragile watermarking is robust to the operation on image which keeps the image content, such as image compression and image enhancement, etc. while it is fragile for the malicious modification. So it is a multipurpose approach.

In the past two decades, since more and more images have been stored in compressed formats such as JPEG and JPEG2000 or transmitted based on vector quantization (VQ) and block truncation coding (BTC), more and more scholars have been engaged in the compressed-domain watermarking schemes. Among them, VQ-based and BTC-based methods are attractive for they are two famous block-based image compression techniques with easy implementation and high efficiency. In the past ten years, several VQ based watermarking approaches [10,11,13,15] have been proposed as a special branch in the digital watermarking area, where the watermark information is embedded in codeword indices. They can be divided into three categories: robust, semi-fragile and multipurpose schemes. The algorithms proposed in [10,11] are robust, the method presented in [13] is semi-fragile, and the scheme provided in [15] is a multipurpose scheme for both copyright protection and content authentication.

In addition to VQ, BTC [17] is another famous block-based lossy image compression technique. It uses a one-bit quantizer to reduce the number of gray levels in each block while preserving the same mean and standard deviation. BTC is a quick, effective and simple block-based lossy image compression technique. It has some characters such as high performance and high channel fault tolerance. And it has a great application value on real-time image transmission. Among several BTC variations, the absolute moment BTC (AMBTC) [18] preserves the first absolute moment instead of the standard deviation along with the mean.

AMBTC is computationally simpler than the original BTC. In the past decade, several watermarking and data hiding schemes for BTC compressed gray-level images have been proposed. The first work was proposed by Lu et al. [12], where the robust watermark is embedded by controlling the VQ-BTC encoding process according to the watermark bits. Later, Lin and Chang [19] put forward a data hiding approach for BTC compressed images by implementing LSB substitution operations on BTC high and low means as well as performing the minimum distortion algorithm on BTC bitplanes. Chuang and Chang [20] presented a hiding algorithm to embed data in the BTC bitplanes of smooth blocks. In addition, Hong et al. [21], Chen et al. [22], and Zhang et al. [23] have proposed several lossless data hiding schemes for BTC-compressed images. Recently, Yang and Lu [14] have proposed a fragile image watermarking scheme whose main idea is to exploit VQ or BTC to encode each block according to the corresponding watermark bit.

The previous BTC-based watermarking or data hiding approaches modify either the BTC encoding process or the BTC-compressed data according to the secret bits, and thus the quality of the watermarked image is the same as or worse than that of the BTC-compressed image. Especially, there are few BTC-based fragile image watermarking schemes that can identify the modified areas and recover the original content. To improve the watermarked image quality and not be detected by the malicious attackers, this paper proposes a new approach that does not really perform the BTC compression on the image but uses the AMBTC quantized data to guide the embedding and extraction processes, and thus lots of data can be embedded as well as a high image quality can be obtained. In addition, the proposed scheme can locate the maliciously modified locations. The rest of this paper is organized as follows. First, Section 2 briefly introduces the AMBTC technique followed by five related works to be compared in the experiments. Section 3 provides a detailed description of the proposed algorithm. Experimental results and a comparison with existing seven works are given in Section 4. Finally, Section 5 summarizes the contributions of our work.

2. Related Work

2.1. Absolute Moment Block Truncation Coding

The absolute moment BTC (AMBTC) [18] is a block-based spatial domain image compression technique. The main idea is to quantize every pixel in an image block into two levels while preserving the mean and the first absolute central moment of a block. The AMBTC method first segments the 256-graylevel input image **X** into non-overlapping small blocks, each being a k-dimensional vector (typically k=4×4), namely, X={x_1, x_2, …, x_N}, where N is the number of blocks. Then, for each block $x_i=(x_{i1}, x_{i2}, \ldots, x_{ik})^T$, the AMBTC scheme separately quantizes each element in x_i into two levels such that the mean value and the first absolute central moment can be preserved in the reconstruction stage. The mean value m_i of each block x_i is taken as the one-bit quantizer's threshold, namely,

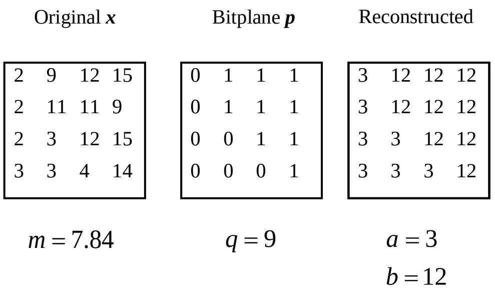
$$m_i = \frac{1}{k}\sum_{j=1}^{k} x_{ij} \tag{1}$$

Let q_i stand for the number of pixels having value not less than m_i. Then the two output quantization level values a_i and b_i can be calculated as follows

$$a_i = \frac{\sum_{j:\, x_{ij}<m_i} x_{ij}}{k-q_i} \tag{2}$$

$$b_i = \frac{\sum_{j:\, x_{ij} \geq m_i} x_{ij}}{q_i} \tag{3}$$

where a_i and b_i are defined as the low mean and the high mean of block $\boldsymbol{x}_i$, respectively. If $q_i=k$, we define $a_i=b_i=m_i$. After the binary quantization is performed for all the pixels in $\boldsymbol{x}_i$, we can obtain a bitplane $\boldsymbol{p}_i$ in which '0' corresponds to the pixels with values less than m_i while '1' corresponds to the rest of the pixels. Thus we can use the triple (a_i, b_i, $\boldsymbol{p}_i$) to describe the compressed version of $\boldsymbol{x}_i$. During the decoding stage, the image block $\boldsymbol{x}_i$ can be easily reconstructed from the bitplane $\boldsymbol{p}_i$ by replacing each '0' with a_i and each '1' with b_i respectively. **Fig. 1** shows an example of encoding and decoding the image block $\boldsymbol{x}$ based on AMBTC. Obviously, for a 256-graylevel 4×4-sized image block, its low mean and high mean are coded separately with 8 bits each, and its bitplane needs 16 bits, so the coding bit rate is (8+8+16)/16=2bpp. Although the coding bit rate is much higher than JPEG and VQ, AMBTC can be performed very fast.



Original $\boldsymbol{x}$

2	9	12	15
2	11	11	9
2	3	12	15
3	3	4	14

$m = 7.84$

Bitplane $\boldsymbol{p}$

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

$q = 9$

Reconstructed

3	12	12	12
3	12	12	12
3	3	12	12
3	3	3	12

$a = 3$
$b = 12$

Fig. 1. An example of encoding a block $\boldsymbol{x}$ by the triple (a, b, p).

2.2. Chuang and Chang's Scheme

For a smooth block $\boldsymbol{x}_i$ whose absolute distance between a_i and b_i is small, its bitplane $\boldsymbol{p}_i$ will be less significant in the AMBTC reconstruction process. Under these circumstances, some suitable locations in $\boldsymbol{p}_i$ may be replaced with the secret bit. Based on this idea, Chuang and Chang's scheme [20] embeds data in the smooth blocks' bitplanes that are determined by the difference between b_i and a_i. If b_i-a_i is not greater than the preset threshold TH, then the block $\boldsymbol{x}_i$ is classified as a smooth block; otherwise, the block is a complex block. Then, a suitable location in the smooth block's bitplane $\boldsymbol{p}_i$ is selected to be replaced with the secret bit. Obviously, the higher the threshold TH is, the more data may be hidden, but the more distortion will be introduced. The extraction process is very simple. From each compressed BTC blocks, the difference b_i-a_i is first calculated, and then whether the difference is less than TH or not is determined. Once confirmed, the secret bit in the bitplane $\boldsymbol{p}_i$ is extracted. One obvious drawback of Chuang and Chang's scheme is that the capacity is determined by the number of smooth blocks.

2.3. Hong et al.'s Scheme

As we know, if we want to exchange a_i and b_i in the compressed triple of block $\boldsymbol{x}_i$, we only need to flip the bitplane $\boldsymbol{p}_i$ into $\overline{\boldsymbol{p}}_i$ in order to obtain the same reconstructed block. Based on this idea, Hong et al.'s embedding scheme [21] can be illustrated as follows: First, all embeddable blocks with $a_i<b_i$ are found, that is to say, the block with $a_i=b_i$ is non-embeddable. Then, for each embeddable block, if the bit to be embedded is ‘1’, then the compressed code is changed from (a_i, b_i, $\boldsymbol{p}_i$) into (b_i, a_i, $\overline{\boldsymbol{p}}_i$); otherwise, do nothing. In other words, the secret bit ‘0’ corresponds to the code (a_i, b_i, $\boldsymbol{p}_i$) and the secret bit ‘1’ corresponds to the code (b_i, a_i, $\overline{\boldsymbol{p}}_i$). The extraction process is very simple. Assume we receive the code (a_i, b_i, $\boldsymbol{p}_i$), we only need to judge the relationship between a_i and b_i. If $a_i > b_i$, then the secret bit is 1; if $a_i<b_i$, the secret bit is 0; otherwise, no secret bit is embedded. Although Hong et al.’s scheme is reversible, it does not consider hiding data in the blocks with $a_i=b_i$.

2.4. Chen et al.’s Scheme

To embed secret data in the blocks with $a_i=b_i$, Chen et al. [22] proposed an improved reversible data hiding algorithm by introducing Chuang and Chang’s bitplane replacement idea to deal with the case $a_i=b_i$ in Hong et al.’s bitplane flipping scheme. The scheme aims to get a payload higher than Hong et al.’s bitmap flipping scheme as well as a stego-image quality improvement over Chuang and Chang’s scheme.

2.5. Yang and Lu’s Method

The above three methods fall into the data hiding category that is a bit different from digital watermarking. Recently, Yang and Lu [14] have proposed a very simple BTC/VQ-domain watermarking scheme. Their embedding procedure can be detailed as follows: the original image $\mathbf{X}$ is first divided into non-overlapping blocks $\boldsymbol{x}_i$, i=1, 2, …, N. For each image block $\boldsymbol{x}_i$, if the watermark bit to be embedded w_i=1, then VQ is used to encode the block $\boldsymbol{x}_i$ by searching for its best-matched codeword $\boldsymbol{c}_j$ in the codebook C. If the watermark bit to be embedded w_i=0, then the AMBTC is used to encode the block $\boldsymbol{x}_i$ by replacing the pixels that are not less than the mean value m_i with the high mean b_i and replacing the pixels that are smaller than the mean value m_i with the low mean a_i. Finally, all encoded image blocks are pieced together to obtain the final watermarked image $\mathbf{X}^{\mathrm{w}}$. The extraction process is just the reverse process of the embedding process as follows: the suspicious image $\mathbf{Y}$ is first divided into non-overlapping blocks $\boldsymbol{y}_i$, i=1, 2, …, N. For each image block $\boldsymbol{y}_i$, it is encoded with VQ and BTC to obtain their corresponding mean squared error $\mathrm{MSE}_{\mathrm{VQ}}$ and $\mathrm{MSE}_{\mathrm{BTC}}$ respectively. If $\mathrm{MSE}_{\mathrm{VQ}} < \mathrm{MSE}_{\mathrm{BTC}}$, then the extracted watermark w_i=1. If $\mathrm{MSE}_{\mathrm{VQ}} > \mathrm{MSE}_{\mathrm{BTC}}$, then the extracted watermark bit w_i=0. If $\mathrm{MSE}_{\mathrm{VQ}} = \mathrm{MSE}_{\mathrm{BTC}}$, then w_i is randomly set to 0 or 1. All obtained watermark bits are pieced together to obtain the final extracted watermark W^{e}.

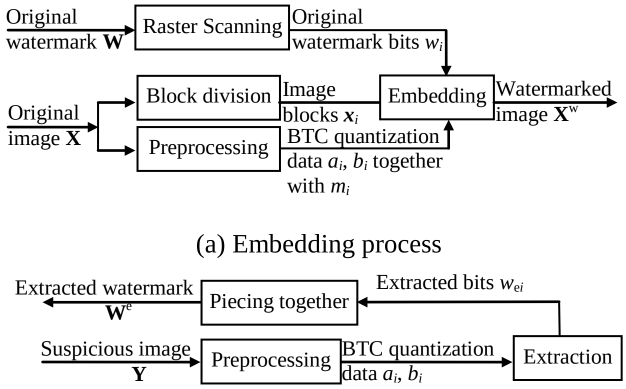
2.6. Zhang et al.’s Scheme

Zhang et al. [23] have proposed an oblivious fragile watermarking scheme for images utilizing edge transitions in BTC bitmaps. In the embedding process of the scheme, the original image was partitioned into non-overlapping 4×4 blocks, then the bitmap of each block was been obtained with BTC. The embedding watermark bit was been used to guide to modify the pixel values until the parity of the number of horizontal edge transitions in the bitmap of the modified block is equal to the embedding watermark bit. Similarly, in the extraction process, only to check the parity of the number of horizontal edge transitions in the BTC bitmap of each non-overlapping 4×4 blocks. Then one watermark bit would be extracted from each block.

3. Proposed Watermarking Scheme

3.1. Preprocessing Stage

From the above discussion, we can see that the existing BTC-based information hiding schemes actually embed the information in the BTC-compressed image. Thus, the watermarked image is usually with a poor quality. In order to improve the image quality, this paper proposes a new train of thought. We still exploit the BTC, but we do not perform the watermark embedding on the BTC compressed image but use the BTC quantization data to guide the watermark embedding and extraction process. The proposed algorithm consists of three processes, namely, the preprocessing stage, the watermark embedding stage and the watermark extraction stage, as shown in **Fig. 2**.



(a) Embedding process

(b) Extraction process

Fig. 2. The block diagram of the proposed watermarking scheme.

From **Fig. 2**, we can see that the preprocessing stage is the common stage for both watermark embedding and extraction processes. The purpose of this stage is to achieve the BTC quantization data as the control parameter. Assume the input image is **X** or **Y** consisting of N blocks and the watermark is a binary image **W** with $2N$ bits. Thus, the preprocessing stage is just a BTC coding process as follows:

Step 1: The input image **X** (or **Y**) is divided into non-overlapping k-dimensional blocks x_i (or y_i), $i = 1, 2, ..., N$.

Step 2: Each image block x_i (or y_i) is encoded by AMBTC, obtaining its mean value m_i and its two quantization levels a_i and b_i.

Step 3: All means are composed of the mean sequence $M = \{m_i \mid i = 1,2,...,N\}$, all low means are composed of the low mean sequence $A = \{a_i \mid i = 1,2,...,N\}$ and all high means consist of the high mean sequence $B = \{b_i \mid i = 1,2,...,N\}$.

3.2. Watermark Embedding Stage

With the means $M = \{m_i \mid i = 1,2,...,N\}$, the low means $A = \{a_i \mid i = 1,2,...,N\}$ and high means $B = \{b_i \mid i = 1,2,...,N\}$ in hand as the guider, now we can describe our watermark embedding process. Before embedding, we perform the raster scanning on the original watermark W, obtaining the watermark bit sequence $W = \{w_1, w_2, ..., w_{2N}\}$. The purpose of our embedding process is to ensure the blind extraction such that the watermark can be extracted from the watermarked image only based on its two quantization values a_i and b_i.

Here, a function $Parity(x)$ is defined as follows:

$$\begin{cases} Parity(x)=1 & \text{if } x \text{ is an odd number} \\ Parity(x)=0 & \text{if } x \text{ is an even number} \end{cases}$$

The central idea is to force the parity of a_i and the parity of b_i to be equal to the two watermark bits. That is, two watermark bits will be embedded in each block. The embedding process can be illustrated detailedly as follows:

Step 1: The original image **X** is divided into non-overlapping k-dimensional blocks x_i, $i = 1,2,...,N$.

Step 2: The embedding process is performed block by block. For each block x_i, the two watermark bits to be embedded are w_{2i-1} and w_{2i}, the embedding procedure is controlled by the quantization data a_i and b_i as following substeps.

Substep 2.1: We count the number of pixels that are not less than m_i and denote it as q_i, thus the number of pixels that are less than m_i is $k - q_i$. Obviously, we have $1 \le q_i \le k$.

Substep 2.2: We find all the pixels in x_i that are not less than m_i, and then arrange them in the descending order, obtaining a 'high' set **H**. Similarly, we find all the pixels in x_i that are less than m_i, and then arrange them in the descending order, obtaining a 'low' set **L**.

Substep 2.3: If both $Parity(a_i) = w_{2i-1}$ and $Parity(b_i) = w_{2i}$ are satisfied, we do not change any pixel in the block x_i, and go to Substep 3. Otherwise, we go to Substep 2.4.

Substep 2.4: We change some pixels in the x_i in order to force the two equations Parity($\hat{a}_i$)=w_{2i-1} and Parity($\hat{b}_i$)=w_{2i} are satisfied for the modified block $\hat{\boldsymbol{x}}_i$. There are two cases:

Case 1: a_i=b_i=m_i(equivalently, q_i =k). In fact, this case corresponds to a block with uniform pixel values. There are three sub-cases as follows.

Sub-case 1.1: Parity(a_i)≠w_{2i-1} but Parity(b_i)=w_{2i}. If a_i>0, we subtract 1 from each of the first half pixels in the block. Otherwise, we add 1 to each of the first half pixels in the block and add 2 to each of the remaining pixels.

Sub-case 1.2: Parity(a_i)=w_{2i-1} but Parity(b_i) ≠w_{2i}. If a_i<255, we add 1 to each of the first half pixels in the block. Otherwise, we subtract 1 from each of the first half pixels in the block and subtract 2 from each of the remaining pixels.

Sub-case 1.3: Parity(a_i) ≠w_{2i-1} and Parity(b_i) ≠w_{2i}. If 0<a_i<255, we subtract 1 from each of the first half pixels in the block and add 1 to each of the remaining pixels. If a_i=0, we add 1 to each of the first half pixels in the block and add 3 to each of the remaining pixels. If a_i=255, we subtract 1 from each of the first half pixels in the block and subtract 3 from each of the remaining pixels.

Case 2: a_i<b_i. There *are* also three sub-cases:

Sub-case 2.1: Parity(a_i)≠w_{2i-1} but Parity(b_i)=w_{2i}. In this case, we perform the following modification rule:

If $\sum_{j:x_{ij}<m_i} x_{ij} \geq k - q_i$, we perform $\hat{x}_{ij} = x_{ij} - 1$ for all $\{x_{ij} \mid x_{ij} > 0, x_{ij} \in \mathbf{L}\}$ repeatedly until $\sum_{j:x_{ij}<m_i} \hat{x}_{ij} = \left(\sum_{j:x_{ij}<m_i} x_{ij} \right) - (k - q_i)$ is satisfied. Thus, we have $\hat{a}_i$ =a_i-1 and therefore $Parity(a_i)$ is changed while $Parity(b_i)$ is still preserved.

Else if $\sum_{j:x_{ij}<m_i} x_{ij} < k - q_i$ and a_i=1, we set $\hat{x}_{ij} = 0$ for all $x_{ij} \in \mathbf{L}$. Thus, we have $\hat{a}_i$ =0=a_i-1, and therefore $Parity(a_i)$ is changed while $Parity(b_i)$ is still preserved.

Else if a_i=0 and b_i>1, we set $\hat{x}_{ij} = 1$ for all $x_{ij} \in \mathbf{L}$. Thus, we have $\hat{a}_i$ =1=a_i+1, and therefore $Parity(a_i)$ is changed while $Parity(b_i)$ is still preserved.

Else if a_i=0 and b_i=1, we set $\hat{x}_{ij} = 1$ for all $x_{ij} \in \mathbf{L}$ and set $\hat{x}_{ij} = x_{ij} + 2$ for all $x_{ij} \in \mathbf{H}$. Thus, we have $\hat{a}_i$ =1= a_i+1 and $\hat{b}_i$ =b_i+2 and therefore $Parity(a_i)$ is changed while $Parity(b_i)$ is still preserved.

Sub-case 2.2: Parity(a_i)=w_{2i-1} but Parity(b_i) ≠w_{2i}. In this case, we perform the following modification rule:

If $\sum_{j:x_{ij}\geq m_i} (255 - x_{ij}) \geq q_i$, we perform $\hat{x}_{ij} = x_{ij} + 1$ for all $\{x_{ij} \mid x_{ij} < 255, x_{ij} \in \mathbf{H}\}$ repeatedly until $\sum_{j:x_{ij}\geq m_i} \hat{x}_{ij} = \left(\sum_{j:x_{ij}\geq m_i} x_{ij} \right) + q_i$. Thus, we have $\hat{b}_i$ =b_i+1 and therefore $Parity(b_i)$ is changed while $Parity(a_i)$ is still preserved.

Else if $\sum_{j:x_{ij}\geq m_i} (255 - x_{ij}) < q_i$ and b_i=254, we set $\hat{x}_{ij} = 255$ for all $x_{ij} \in \mathbf{H}$. Thus, we have $\hat{b}_i$ =255=b_i+1 and therefore $Parity(b_i)$ is changed while $Parity(a_i)$ is still preserved.

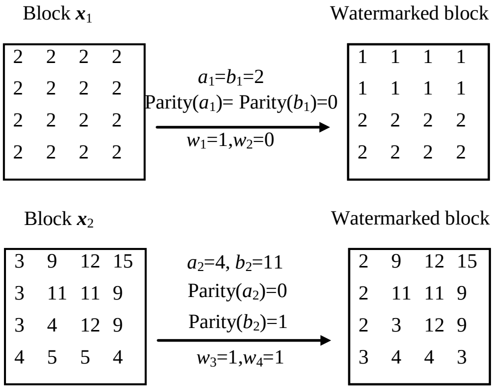
Else if b_i=255 and a_i<254, we set $\hat{x}_{ij} = 254$ for all $x_{ij} \in \mathbf{H}$. Thus, we have $\hat{b}_i$ =254=b_i-1 and therefore $Parity(b_i)$ is changed while $Parity(a_i)$ is still preserved.

Else if b_i=255 and a_i=254, we set $\hat{x}_{ij} = 254$ for all $x_{ij} \in \mathbf{H}$ and set $\hat{x}_{ij} = x_{ij} - 2$ for all $x_{ij} \in \mathbf{L}$. Thus, we have $\hat{b}_i$ =254=b_i-1 and $\hat{a}_i$ =a_i-2 and therefore $Parity(b_i)$ is changed while $Parity(a_i)$ is still preserved.

Sub-case 2.3: $Parity(a_i) \neq w_{2i-1}$ and $Parity(b_i) \neq w_{2i}$. In this case, the above two rules of Sub-case 2.1 and Sub-case 2.2 will be performed.

Step 3: If every block has been embedded with 2 watermark bits, then the algorithm is terminated with the watermarked image $\mathbf{X}^{w}$. Otherwise, go to Substep 2.1 for next block.

In order to understand the proposed embedding process more clearly, two concrete examples are described in **Fig. 3**. For the uniform block x_1, assume the two watermark bits to be embedded are w_1=1 and w_2=0, since a_1=b_1=2, thus Parity(a_1)≠w_1 and Parity(b_1)=w_2, and it falls into Sub-case 1.1. For block $\boldsymbol{x}_2$, assume the two watermark bits to be embedded are w_3=1 and w_4=1, since a_2=4 and b_2=11, thus Parity(a_2)≠w_3 and Parity(b_2) =w_4, and it falls into Sub-case 2.1.



$$\text{Block } \boldsymbol{x}_1: \begin{array}{|cccc|}\hline 2 & 2 & 2 & 2\\ 2 & 2 & 2 & 2\\ 2 & 2 & 2 & 2\\ 2 & 2 & 2 & 2\\\hline\end{array} \xrightarrow[w_1=1, w_2=0]{a_1=b_1=2 \quad \text{Parity}(a_1)=\text{Parity}(b_1)=0} \text{Watermarked block}: \begin{array}{|cccc|}\hline 1 & 1 & 1 & 1\\ 1 & 1 & 1 & 1\\ 2 & 2 & 2 & 2\\ 2 & 2 & 2 & 2\\\hline\end{array}$$

$$\text{Block } \boldsymbol{x}_2: \begin{array}{|cccc|}\hline 3 & 9 & 12 & 15\\ 3 & 11 & 11 & 9\\ 3 & 4 & 12 & 9\\ 4 & 5 & 5 & 4\\\hline\end{array} \xrightarrow[w_3=1, w_4=1]{a_2=4,\ b_2=11 \quad \text{Parity}(a_2)=0 \quad \text{Parity}(b_2)=1} \text{Watermarked block}: \begin{array}{|cccc|}\hline 2 & 9 & 12 & 15\\ 2 & 11 & 11 & 9\\ 2 & 3 & 12 & 9\\ 3 & 4 & 4 & 3\\\hline\end{array}$$

Fig. 3. Concrete embedding examples.

3.3. Watermark Extraction Stage

According to **Fig. 2** (b), we can see that our watermark extraction process is a blind process such that we do not require the original image during the extraction process. It is very simple because we can get the two watermark bits from each block only based on checking the parities of a_i and b_i. Before extraction, the same preprocessing step is performed on the suspicious image **Y** to obtain its low mean sequence $\mathbf{A}=\{a_i \mid i=1,2,\ldots,N\}$ and high mean sequence $\mathbf{B}=\{b_i \mid i=1,2,\ldots,N\}$. For each block $\boldsymbol{y}_i$, if a_i is odd, then we can extract the watermark bit $w_{2i\text{-}1}$=1. Otherwise, we can extract the watermark bit $w_{2i\text{-}1}$=0. Similarly, if b_i is odd, then we can extract the watermark bit w_{2i}=1. Otherwise, we can extract the watermark bit w_{2i}=0. After all blocks are performed, we piece these bits together to obtain the final extracted watermark $\mathbf{W}^{e}=\{w_{e1}, w_{e2},\ldots, w_{e2N}\}$.

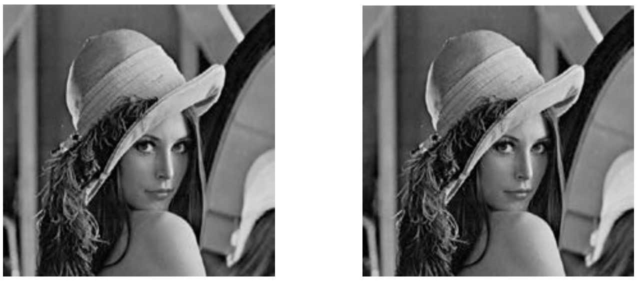
4. Experimental Results and Analysis

4.1. Performance Testing

To evaluate the performance of the proposed watermarking scheme, the 256 grayscale 512×512 sized Lena image is used for watermarking. The Lena image is divided into 16384 blocks of size 4×4. A binary image of size 256×128 serves as the watermark W for embedding. **Fig. 4**(a) shows the original Lena image and **Fig. 4**(b) shows the watermarked Lena image with PSNR=51.11dB. To check the fragility of our algorithm, we perform several attacks on the watermarked image, including brightness enhancement by 10%, contrast enhancement by 10%, JPEG compression with QF=100%, JPEG compression with QF=80%, image cropping in the middle part of the image, blurring with a 3×3 window and a threshold 25, median filtering with a 3×3 window, and rotation by 1°. The attacked images are shown in **Fig.**

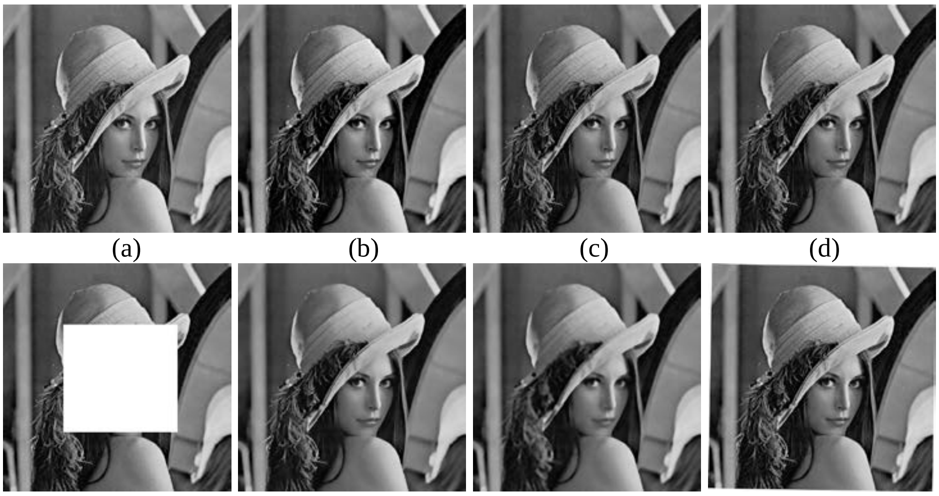
5(a)-(h). The corresponding extracted watermarks are shown in Fig. 6(c)-(j), while the original watermark is shown in Fig. 6(a) and the extracted watermark under no attacks is shown in Fig. 6(b). The performance of extracted watermarks are evaluated by normalized cross-correlation (NC). From these results we can see that the proposed algorithm is fragile to most attacks except for the change in brightness which is caused by the AMBTC's special ability in preserving the mean value and the first absolute central moment. In that the texture of the image belongs to the part of the ownership and the operation on brightness enhancement can not change the texture of the image, so our proposed scheme can be used for texture authentication. From Fig. 5(e) and Fig. 6(g), we can see that our scheme can locate the maliciously modified locations. In addition, the proposed algorithm is blind because the original image is not required during the extraction process.

In addition, we have two methods to improve the security of our proposed scheme. One method is to keep the size of the non-overlapping k-dimensional blocks as a key. The other method is to select the different position from the image as the first block's left-top pixel, and the different position is a key.



(a) Original cover image (b) Watermarked image

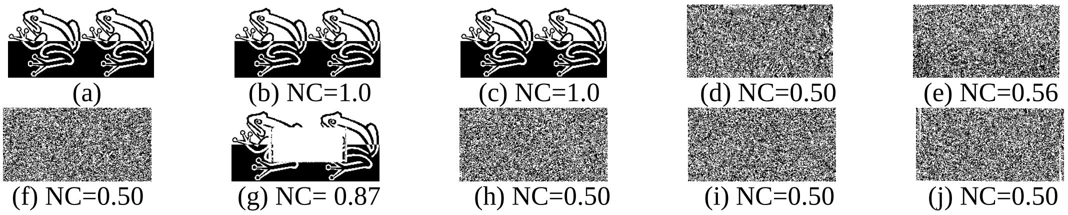
Fig. 4. The cover and watermarked images.



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 5. The attacked watermarked images under 8 attacks, namely, brightness enhancement, contrast enhancement, JPEG (QF=100), JPEG (QF=80), cropping, blurring, median filtering, and rotation, respectively.



(a) (b) NC=1.0 (c) NC=1.0 (d) NC=0.50 (e) NC=0.56

(f) NC=0.50 (g) NC= 0.87 (h) NC=0.50 (i) NC=0.50 (j) NC=0.50

Fig. 6. The original watermark, the extracted watermark from unattacked watermarked image and extracted watermarks under various attacks, namely, brightness enhancement, contrast enhancement, JPEG (QF=100), JPEG (QF=80), cropping, blurring, median filtering, and rotation, respectively.

4.2. Performance Comparison

To show the superiority of our proposed algorithm, we use six test images, Lena, Peppers, Mandrill, Boat, Goldhill and Jet_F16, of the same size 512×512 with 256 grayscales, as shown in **Fig. 7**. Comparisons among our algorithm, Chuang and Chang’s algorithm, Hong et al.’s algorithm, Chen et al.’s scheme, Yang and Lu’s Method and Zhang et al.’s scheme which are described in Section 2 are performed under the same block size 4×4. Two aspects of performance are considered, namely, the peak signal to noise ratio (PSNR) representing the quality of the watermarked image, and the capacity representing the maximum number of secret bits that can be hidden. Here, although the capacity is not a very important index for a watermarking scheme, but we would like to demonstrate that our scheme can hide large number of bits while preserving high image quality compared to existing BTC-based schemes.

As shown in **Table 1**, the PSNRs of watermarked images based on our scheme are much higher than those of the existing schemes. With respect to the embedding capacity, compared with existing schemes, our scheme can embed 2bits per block, which is at least double capacity of the existing schemes. Taking the above two attributes into comprehensive consideration, the proposed scheme is a better method for its high capacity and high PSNR.

Fig. 7. Six test images.

Table 1. Comparisons between the proposed and existing BTC-based hiding schemes (k=4×4)

Algorithm	**Performance**	**Lena**	**Peppers**	**Mandrill**	**Boat**	**Goldhill**	**Jet_F16**
The Proposed Scheme	PSNR(dB)	51.11	51.01	51.12	51.13	51.10	51.17
	Capacity(Bits)	32768	32768	32768	32768	32768	32768
Chuang and Chang’s	PSNR(dB)	31.66	31.23	25.91	30.79	32.53	30.85
	Capacity(Bits)	13048	13464	5168	11981	12487	12631
Hong et al.’s	PSNR(dB)	32.03	31.60	26.00	31.18	33.16	31.03
	Capacity(Bits)	16384	16099	16384	16384	16384	16384
Chen et al.’s	PSNR(dB)	32.03	31.60	26.00	31.18	33.16	31.03
	Capacity(Bits)	16384	20944	16384	16384	16384	16384
Yang and Lu’s	PSNR(dB)	32.05	29.44	25.13	31.00	32.71	30.24
	Capacity(Bits)	16384	16384	16384	16384	16384	16384
Zhang et al.’s	PSNR(dB)	38.63	38.08	35.41	38.02	40.72	36.46
	Capacity(Bits)	16384	16384	16384	16384	16384	16384

We also compare our scheme with two classical LSB-based schemes, one is proposed by Mielikainen [24], the other is proposed by Zhang and Wang [25]. In Mielikainen’s scheme, he proposed a modified LSB matching-based steganographic method for embedding message bits into a still image. The embedding operation is performed on every two pixels. The LSB of the first pixel carries one bit of information, and another bit of information is presented by a function of the two pixel values. Their method made fewer changes to the cover images comparing to LSB matching and improved the image quality. However, there was a problem to handle with the saturated pixels, i.e., the pixels with value ‘0’ or ‘255’ in the 256-grayscale image. That is, the number of non-saturated pixels is the capacity of this scheme. In Zhang and Wang’s scheme [25], a secret digital in (2n+1)-ary notational system is embedded into an n pixels group, in which only one pixel is increased or decreased by 1. In our experiment, we set n=54, then we have the fixed capacity 32856(=ln109/ln2*512*512/54). The comparison results are shown in **Table 2**. From this table, we can see that the two LSB-based schemes have better performance than our scheme in terms of PSNR and capacity, for they are pixel-based method and only modify the LSB of the pixel to be embedded, while our scheme is block-based and we modify more than one pixel in each block. However, our scheme is a semi-fragile scheme that is robust to brightness or contrast changes, thus our scheme has some special characteristics that the LSB-based schemes do not have. In addition, we provide a new watermarking mechanism of pixel-domain embedding followed by compressed-domain analysis and extraction.

Table 2. Comparisons between the proposed and recent two LSB-based watermarking schemes

Algorithm	Performance	Lena	Peppers	Mandrill	Boat	Goldhill	Jet_F16
Proposed	PSNR(dB)	51.11	51.01	51.12	51.13	51.10	51.17
	Capacity(Bits)	32768	32768	32768	32768	32768	32768
Mielikainen’s	PSNR(dB)	52.40	52.69	52.3912	52.40	52.40	52.41
	Capacity(Bits)	262144	244586	262078	262126	262144	261120
Zhang and Wang’s	PSNR(dB)	65.46	65.46	65.46	65.46	65.46	65.46
	Capacity(Bits)	32856	32856	32856	32856	32856	32856

5. Conclusion

Considering the existing BTC-based watermarking or data hiding schemes obtain low-quality images or have less hiding capacity, in this paper, we propose a new blind fragile image watermarking approach by exploiting the AMBTC technique to guide the watermark embedding and extraction processes. It is a simple but efficient method suitable for texture authentication and data hiding. The central idea is to force the parity of the low mean and high mean of each block to be equal to the two watermark bits to be embedded respectively. Another contribution of this paper is that we have developed a semi-fragile watermarking scheme with the mechanism of pixel-domain-based embedding and compressed-domain-based extraction, which can make full use of the statistical characteristics of the compressed-domain. Experimental results show that it is fragile to most kinds of attacks and it is better than some existing BTC-based image watermarking schemes. The future work will be concentrated on how to further improve the capacity while maintaining the high image quality and how to recover the maliciously modified blocks.

References

[1] X. M. Niu, Z. M. Lu and S. H. Sun, "Digital watermarking of still images with gray-level digital watermarks," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 1, pp. 137-145, February, 2000. Article (CrossRef Link)

[2] X. M. Niu, Z. M. Lu and S. H. Sun, "Digital image watermarking based on multiresolution decomposition," *Electronics Letters*, vol. 36, no. 13, pp. 1108-1110, June, 2000. Article (CrossRef Link)

[3] H. Luo, F. X. Yu, Z. L. Huang and Z. M. Lu, "Blind image watermarking based on discrete fractional random transform and subsampling," *International Journal for Light and Electron Optics*, vol. 122, no. 4, pp. 311-316, February, 2011. Article (CrossRef Link)

[4] H. X. Wang, Z. M. Lu, Y. N. Li and S. H. Sun, "A compressed domain multipurpose video watermarking algorithm using vector quantization," *International Journal of Innovative Computing, Information and Control*, vol. 5, no. 5, pp. 1441-1450, May, 2009. Article (CrossRef Link)

[5] Y. G. Wang, Z. M. Lu, F. Liang and Y. Zheng, "Robust dual watermarking algorithm for AVS video," *Signal Processing: Image Communication*, vol. 24, no. 4, pp. 333-344, April, 2009. Article (CrossRef Link)

[6] H. X. Wang, Z. M. Lu, J. S. Pan and S. H. Sun, "Robust blind video watermarking with adaptive embedding mechanism," *International Journal of Innovative Computing, Information and Control*, vol. 1, no. 2, pp. 247-259, June, 2005. Article (CrossRef Link)

[7] C. H. Wu, Y. Zheng, W. H. Ip, C. Y. Chan, K. L. Yung and Z. M. Lu, "A flexible H.264/AVC compressed video watermarking scheme using particle swarm optimization based dither modulation," *AEU - International Journal of Electronics and Communications*, vol. 65, no.1, pp. 27-36, January, 2011. Article (CrossRef Link)

[8] Z. M. Lu, B. Yan and S. H. Sun, "Watermarking combined with CELP speech coding for authentication," *IEICE Transactions on Information and Systems*, vol. E88-D, no. 2, pp. 330-334, February, 2005. Article (CrossRef Link)

[9] B. Yan, Z. M. Lu and S. H. Sun, "Security of autoregressive speech watermarking model under guessing attack," *IEEE Trans Information Forensics and Security*, vol. 1, no. 3, pp. 386-390, September, 2006. Article (CrossRef Link)

[10] Z. M. Lu, J. S. Pan and S. H. Sun, "VQ-based digital image watermarking method," *Electronics Letters*, vol. 36, no. 14, pp. 1201-1202, July, 2000. Article (CrossRef Link)

[11] Z. M. Lu and S. H. Sun, "Digital image watermarking technique based on vector quantization," *Electronics Letters*, 2000, vol. 36, no. 4, pp. 303-305, February, 2000. Article (CrossRef Link)

[12] Z. M. Lu, C. H. Liu and S. H. Sun, “Digital image watermarking technique based on block truncation coding with vector quantization,” *Chinese Journal of Electronics*, vol. 11, no. 2, pp.152-157, February, 2002. Article (CrossRef Link)

[13] Z. M. Lu, C. H. Liu, D. G. Xu and S. H. Sun, “Semi-fragile image watermarking method based on index constrained vector quantization,” *Electronics Letters*, vol. 39, no. 1, pp. 35-36, January, 2003. Article (CrossRef Link)

[14] C. N. Yang and Z. M. Lu, “Blind fragile image watermarking based on vector quantization and block truncation coding,” *ICIC Express Letters Part B: Applications*, vol. 2, no. 4, pp.905-910, August, 2011. Article (CrossRef Link)

[15] Z. M. Lu, D. G. Xu and S. H. Sun, “Multipurpose image watermarking algorithm based on multistage vector quantization,” *IEEE Transactions on Image Processing*, vol. 14, no. 6, pp. 822-831, June, 2005. Article (CrossRef Link)

[16] Z. M. Lu, C. H. Liu and H. Wang, “Image retrieval and content integrity verification based on multipurpose image watermarking scheme,” *International Journal of Innovative Computing, Information and Control*, vol. 3, no. 3, pp. 621-630, June, 2007. Article (CrossRef Link)

[17] E. J. Delp and O. R. Mitchell, “Image compression using block truncation coding,” *IEEE Transactions on Communications*, vol. 27, no. 9, pp. 1335-1342, September, 1979. Article (CrossRef Link)

[18] M. D. Lema and O. R. Mitchell, “Absolute moment block truncation coding and its application to color images,” *IEEE Transactions on Communications*, vol. 32, no. 10, pp. 1148-1157, October, 1984. Article (CrossRef Link)

[19] M. H. Lin and C. C. Chang, “A novel information hiding scheme based on BTC,” in *Proc. of 4th Int. Conf. Computer and Information Technology*, pp. 66-71, September 14-16, 2004. Article (CrossRef Link)

[20] J. C. Chuang and C. C. Chang, “Using a simple and fast image compression algorithm to hide secret information,” *International Journal of Computers and Applications*, vol. 28, no. 4, pp. 329-333, October, 2006. Article (CrossRef Link)

[21] W. Hong, T. S. Chen and C. W. Shiu, “Lossless steganography for AMBTC compressed images,” in *Proc. of 1st Int. Congress on Image and Signal Processing*, vol. 2, pp. 13-17, May 27-30, 2008. Article (CrossRef Link)

[22] J. Chen, W. Hong, T. S. Chen and C. W. Shiu, “Steganography for BTC compressed images using no distortion technique,” *The Imaging Science Journal*, vol. 58, no. 4, pp. 177-185, August, 2010. Article (CrossRef Link)

[23] Y. Zhang, Z. M. Lu, and D. N. Zhao, “An oblivious fragile watermarking scheme for images utilizing edge transitions in BTC bitmaps,” *Sci China Inf Sci*, vol. 55 no. 11, pp. 2570–2581, November, 2012. Article (CrossRef Link)

[24] J. Mielikainen, “LSB matching revisited,” *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 285-287, May, 2006. Article (CrossRef Link)

[25] X. Zhang and S. Wang, “Efficient steganographic embedding by exploiting modification direction,” *IEEE Communications Letters*, vol. 10, no. 11, pp.781-783, November, 2006. Article (CrossRef Link)

Dongning Zhao received her B.S. and M.S. in communication engineering from Nanjing Institute of Communication Engineering, Nanjing, China, in 2001 and 2004, respectively. She is currently working toward the Ph.D. degree with ATR National Defense Technology Key Laboratory, College of Information Engineering, Shenzhen University. Her research interests include multimedia security, information hiding and digital watermark.

Weixin Xie received his B.S. from Xi'an Military Telecommunication Engineering Institute, Xi'an, China, in 1965. From 1981 to 1983, he was a visiting scholar with University of Pennsylvania, Philadelphia, USA. And from 1989 to 1990, he was a visiting professor with University of Pennsylvania, Philadelphia, USA. He is currently a professor with ATR National Defense Technology Key Laboratory, College of Information Engineering, Shenzhen University. He has published more than 100 papers in the journals. His research interests include intelligent information processing and sensor network.