A novel reversible image authentication scheme for digital images

Chun-Chi Lo, Yu-Chen Hu

Article info
Article history:
Received 20 August 2013
Received in revised form
28 October 2013
Accepted 24 November 2013
Available online 1 December 2013

Keywords:
Tamper detection
Image authentication
Random number generator
Digital signature
Fragile watermark

Abstract
Image authentication is to protect the image integrity when the digital image is illegally modified. Most of the image authentication schemes proposed so far employed the irreversible data hiding approach to embed the authentication data into the cover images. In this paper, a novel reversible image authentication scheme for digital images is proposed. In the proposed scheme, the authentication codes are generated by using the random number values induced by the selected random number seed. Then, the authentication codes are embedded into the cover image. Experimental results show that the proposed scheme achieves good detecting accuracy while keeping good image quality of the embedded image. The proposed reversible image authentication can be employed to protect the image integrity of the general-purposed images as well as the special-purposed images such as the medical and military images.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

In recent decades, the research issue towards the image integrity protection becomes more and more important because the digital images can be easily copied and modified by using the image processing software such as Adobe PhotoShop and PhotoImpact. The traditional cryptography approach can only be used to protect the security of digital images. When one cryptography scheme, such as RSA, AES, MD5, or RSA, is employed to process the digital image, any change from the encrypted data can be detected. However, the tampered areas in the image cannot be located by using the cryptography scheme. In other words, the cryptography approach is not suitable for the image integrity protection.

The research toward image authentication [1] for image integrity protection had thus been proposed. In general, the image authentication schemes can be classified into two main categories: signature-based schemes [2–5] and fragile watermark-based schemes [6–16]. In a signature-based scheme, the given image is processed by using the hash function and the hashed result is encrypted by using the public key cryptosystem to generate the digital signature. Then, the digital signature of the image to be protected is stored in a trust third party. When the given image is to be authenticated, the digital signature is extracted from the trust third party and it is compared to the other signature that is generated from the image to detect the tampered areas.

In a fragile watermark-based scheme, the watermark data is embedded into the cover image. Typically, the watermark data is generated by using either the image features extracted from the cover image or the random values induced by the selected random number seed. When the desired image is to be authenticated, the watermark data is extracted from the image to detect the tampered areas.
areas. The detection accuracy and the image quality of the embedded image are the two major considerations for the fragile watermark-based image authentication approach.

In 2001, Lin and Chang proposed a semi-fragile watermarking scheme [6] which can be resistant to lossy compression for JPEG images. Wong and Memon proposed the secret and public key image watermarking schemes [7] for image authentication and ownership verification in 2001. Lee and Lin proposed the dual watermark for both image tamper detection and image recovery [8] in 2008. Ahmed and Siyal proposed an image authentication scheme [9] based on the hash function in 2010. Qi and Xin proposed a quantization-based semi-fragile watermarking scheme [10] for content authentication in 2011. Chan proposed the image authentication method by applying the Hamming code to the rearranged bits in 2011 [11]. A novel watermarking scheme with flexible self-recovery quality had been proposed in 2011 [12]. In this scheme, the embedded watermark data for content recovery are computed from the original discrete cosine transform coefficients of the host image. When a part of a watermarked image is tampered, the watermark data in the area without any modification can be extracted. In addition, a novel fragile watermarking scheme with content restoration capability had been proposed in 2012 [13]. In this scheme, authentication bits are generated using the image hashing method with a folding operation. The low-frequency component of the non-subsampled contourlet transform coefficients is used to encode the restoration bits for each block by adaptive bit allocation mechanism.

The fragile watermarking schemes [6–13] had been proposed to protect the image integrity of the digital images in raw format. In addition to the raw images, the compressed images of vector quantization and block truncation coding had been studied to design the image authentication schemes. Chung and Hu proposed an adaptive image authentication scheme for the compressed images of vector quantization in 2011 [14]. In this scheme, the authentication codes are generated by the random values induced by the selected random seed. Then, the authentication codes are embedded into the indices of the compressed images of vector quantization. In 2013, a tamper detection scheme [15] for block truncation coding had been proposed. In this scheme, 1-bit authentication code of each image block is generated from the quantization levels. Multiple copies of the authentication data are embedded into the bit maps of compressed image blocks based on the block permutations. These block permutations are generated by using the random sequences induced by the selected random number seeds. In 2013, a joint image coding and image authentication scheme for block truncation coding had been introduced [16]. In this scheme, the digital image in raw format is compressed by block truncation coding. While the image coding process is executed, the authentication code of each image block is embedded into the compressed codes to protect the image integrity.

In addition to the signature-based schemes and the fragile watermark-based schemes for image authentication mentioned above, a robust image hashing scheme using ring-based entropies had been proposed [17] for image authentication. In this scheme, the image hashing is achieved by converting the input image into a normalized image. The normalized image is then divided into different rings and the ring-based entropies are extracted to produce a hash.

In general, data hiding schemes can be classified into two categories: irreversible data hiding schemes and reversible data hiding schemes [18]. In the irreversible data hiding schemes, secret data is first embedded into the original cover images to generate the stego-images. Then, secret data can be extracted when it is needed by executing the secret extraction procedure. However, the original cover images cannot be restored.

The reversible data hiding schemes are also called the lossless data hiding schemes. The reversible data hiding schemes can extract the secret data and recover the original cover images simultaneously. Two main approaches of the reversible data hiding schemes had been introduced. They are the histogram shifting approach [19–22] and the difference expansion approach [23–25]. Some reversible data hiding schemes [26–28] that use the mix of these two approaches had also been proposed.

Ni et al. proposed the histogram shifting scheme (HS) [19] for reversible data hiding in 2006. In this scheme, the occurrences of all possible pixel values in the cover image are calculated to generate the image histogram. The required pairs of peak and zero points from the image histogram are searched. Secret data is embedded into the pixels in the peak point. Those pixels whose values ranges from the peak point to the zero point have to be modified. By contrast, those pixels whose values are out of the range are not changed. The hiding capacity of HS equals the number of pixels in the peak point. The larger the number of pixels in the peak point is, the higher hiding capacity the scheme has. To increase the hiding capacity of HS, more pairs of peak and zero points can be used. Sometimes, it is impossible to find out more pairs of peak and zero points because the zero points are not searched.

Tsai et al. proposed a prediction-based histogram shifting scheme (PBHS) [20] in 2009. The goal of PBHS is to improve the hiding capacity of the HS scheme. In PBHS, the host image is partitioned into non-overlapped image blocks. The center pixel in each block is selected as the basic pixel for predictive coding. For any block, the difference between each pixel and the basic pixel is calculated to generate the prediction error. No prediction error is generated for the basic pixel. After processing all image blocks, the residual image is generated. Then, the residual image histogram can be produced. Finally, the residual image histogram is employed to embed the secret data. Similarly, multiple pairs of peak and zero points can be selected to embed secret data into the host image.

For a reversible data hiding scheme based on histogram shifting, overflow and underflow problems are unavoidable. For example, it is not allowed to modify the pixel whose value is 0 or 255 in the HS scheme. To solve the overflow and underflow problems, the location map that records the overflow and underflow pixels can be generated. The location map can be viewed as the overhead of the
histogram shifting scheme. The storage cost of the location map can be reduced by losslessly encoding it by any lossless data compression technique and then embed the compressed data into the host image.

In addition to the location map method, one possible solution to solve the overflow and underflow problems is described here. If the number of the overflow and underflow pixels is quite small, the positions of these pixels can be recorded. For example, 18 bits are needed to record the position of one pixel in the grayscale image of $512 \times 512$ pixels. The position information can then be embedded into the host image.

From the literature concerning the fragile watermark-based image authentication schemes, we find that the irreversible data hiding approach is employed in most schemes to embed the authentication data into the cover images. As far as we know, the irreversible data hiding approach provides a higher hiding capacity than the reversible data hiding approach. In other words, the design of the irreversible image authentication scheme is much easier than that of the reversible image authentication because more authentication data can be embedded into the cover image. However, the irreversible image authentication approach is not suitable for image integrity protection of some important images, such as the remote sensing images, the medical images, the military images, because any alterations of these images are not permitted.

To protect the image integrity for the general-purposed images as well as the special-purposed images, a reversible image authentication scheme for digital images is proposed in this paper. This scheme is a kind of the fragile watermark-based image authentication scheme. A block-based data embedding process is designed and the authentication data of each image block is embedded into the embeddable positions based on the prediction-based histogram shifting. The rest of this paper is organized as follows. Section 2 will present the proposed scheme. The experimental results are discussed in Section 3. Finally, the conclusions is given in Section 4.

2. The proposed scheme

The goal of the proposed scheme is to detect the tampered areas for the digital images. If the image is not tampered, the authentication data can be extracted and the embedded image can be recovered back to the original image. If the image is tampered, the tamper areas of the image will be detected. The proposed scheme consists of three procedures: authentication codes generation procedure, the authentication codes embedding procedure, and the tamper detection procedure which are described in the following subsections.

2.1. The authentication codes generation procedure

Suppose the grayscale image of $W \times H$ pixels is to be processed. It is divided into non-overlapping image blocks of $n \times n$ pixels. A total of $w \times h$ image blocks are divided where $w=W/n$ and $h=H/n$. A total number of $w \times h$ authentication codes will be generated. To generate the authentication codes, the pseudo random number generator (PRNG) with a predefined seed $S$ is used to generate $w \times h$ random values. Each random value $v$ is then converted to the 1-bit authentication code $(c)$ by using the following equation:

$$c = v \mod 2.$$  \hfill (1)

As we know, the hiding capacities of the reversible data hiding schemes are generally much less than those of the irreversible data hiding schemes. Therefore, only 1-bit authentication code is generated for each image block.

2.2. The authentication codes embedding procedure

Basically, the authentication codes embedding procedure is similar to the data embedding procedure in the PBHS scheme. The difference between them is to embed the authentication codes or secret data into the embeddable positions in the residual image. In the PBHS scheme,
1-bit secret data is embedded into each embeddable residual value, but 1-bit authentication code is embedded into the embeddable locations of each image block in the proposed authentication codes embedding procedure. The flowchart of authentication codes embedding procedure is depicted in Fig. 1. In the proposed scheme, the cover image is first processed by the predictive coding process. In this process, the cover image is divided into

![Fig. 3. Four test cases for tamper refinement. (a) Case 1, (b) Case 2, (c) Case 3 and (d) Case 4.](image)

![Fig. 4. Six testing images of 512 × 512 pixels. (a) Airplane; (b) boat; (c) girl; (d) goldhill; (e) lenna; and (f) pepper.](image)

<table>
<thead>
<tr>
<th>Images</th>
<th>$p=1$</th>
<th>$p=2$</th>
<th>$p=3$</th>
<th>$p=4$</th>
<th>$p=5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airplane</td>
<td>36,532</td>
<td>63,616</td>
<td>63,620</td>
<td>63,623</td>
<td>63,625</td>
</tr>
<tr>
<td>Boat</td>
<td>23,047</td>
<td>44,631</td>
<td>44,637</td>
<td>44,641</td>
<td>44,644</td>
</tr>
<tr>
<td>Girl</td>
<td>22,442</td>
<td>43,746</td>
<td>43,767</td>
<td>43,767</td>
<td>43,782</td>
</tr>
<tr>
<td>Goldhill</td>
<td>16,463</td>
<td>31,903</td>
<td>31,909</td>
<td>31,911</td>
<td>31,913</td>
</tr>
<tr>
<td>Lenna</td>
<td>24,447</td>
<td>46,009</td>
<td>46,014</td>
<td>46,018</td>
<td>46,022</td>
</tr>
<tr>
<td>Pepper</td>
<td>22,345</td>
<td>42,960</td>
<td>42,966</td>
<td>42,969</td>
<td>42,971</td>
</tr>
<tr>
<td>Average</td>
<td>24,212.667</td>
<td>45,477.500</td>
<td>45,485.500</td>
<td>45,488.167</td>
<td>45,492.833</td>
</tr>
</tbody>
</table>
non-overlapping image blocks of \( n \times n \) pixels. The center pixel in each block is chosen as the basic pixel for predictive coding. For any block, the difference between each pixel and the basic pixel is computed to generate the prediction error. After processing all image blocks, the residual image is generated. The occurrences of all possible values in the residual image are calculated to generate the residual image histogram.

Let \( p \) denote the pair of peak and zero points that will be used to embed the authentication data. The value of \( p \) in the residual histogram are searched before the actually data embedding process is executed. A block-based data embedding process that is similar to that in the PBHS scheme is employed here. In the PBHS scheme, the secret data is randomly generated and each embeddable residual value is used to carry 1-bit secret data.

In the proposed authentication codes embedding process, the residual values are divided into non-overlapping image blocks. Let \( c \) denote the 1-bit authentication code that will be embedded into the embeddable residual values in each residual block. The same authentication code \( c \) is embedded into these embeddable residual values in each residual block. By sequentially embedding the 1-bit authentication code of each block in the same way, the embedded residual values are generated. Finally, the stego-image is produced by performing the inverse predictive coding process on the embedded residual values.

**Fig. 5.** Image qualities of the embedded images of the proposed scheme.

**Fig. 6.** Embedded images “Lenna” of the proposed scheme when the block size equals 4 \( \times \) 4. (a) \( p = 1 \) with 51.454 dB; (b) \( p = 2 \) with 48.826 dB; (c) \( p = 3 \) with 48.826 dB; and (d) \( p = 4 \) with 48.826 dB.
Recall that \((n \times n - 1)\) residual values are generated in the predictive coding process when image size equals \(n \times n\). The total number of embeddable residual values is ranged between 0 and \((n \times n - 1)\). It may happen that there is no embeddable residual value in some image blocks. In other words, no authentication codes can be embedded.

Fig. 7. The tampered object of the test. (a) The tampered object of 6217 pixels and (b) the binary tampered object.

Fig. 8. Tampered images “Lenna” of the proposed scheme when the block size equals \(4 \times 4\). (a) \(p = 1\); (b) \(p = 2\); (c) \(p = 3\); and (d) \(p = 4\).
Fig. 9. The difference images for the tamper test. (a) Pixel difference image ($p=1$); (b) block difference image ($p=1$); (c) pixel difference image ($p=2$); (d) block difference image ($p=2$); (e) pixel difference image ($p=3$); (f) block difference image ($p=3$); (g) pixel difference image ($p=4$); and (h) block difference image ($p=4$).
into some image blocks where their residual values are not located at the peak points.

2.3. The tamper detection procedure

The goal of the tamper detection procedure is to detect whether the given image of \( W \times H \) pixels is modified or not. If this image is modified, the tamper areas of the image will be detected. If this image is not tampered, the authentication codes can be extracted and the original cover image can be recovered. Some system parameters, such as \( W, H, p, n, \) and the random number seed \( S \), should be known in advance. In addition, \( p \) pairs of peak and zero points should be available.

The flowchart of the tamper detection procedure is shown in Fig. 2. To determine whether these \( w \times h \) image blocks are tampered or not, two sets of authentication codes of the image block are to be generated. The PRNG with the random seed \( S \) is employed to generate \( w \times h \) random values. Each random value \( v \) is then converted to 1-bit authentication code \( c \) by using Eq. (1). By collecting all the authentication codes of the image blocks, the first set of authentication codes is generated.

The second set of the authentication codes will be extracted by using the data extraction procedure of the PBHS scheme. To extract the authentication codes, the predictive coding is performed on the given image to generate the residual image. Then, the residual histogram is produced. The authentication codes are then extracted by processing the residual values with the use of \( p \) pairs of peak and zero point. The residual values are recovered to generate the residual image. Then, the cover image is rebuilt by performing inverse predictive coding.

To generate the second set of the authentication codes, the authentication codes that are extracted by using the PBHS data extraction procedure are processed. All the authentication codes in each block are collected together.

Table 1 lists the hiding capacities of the proposed scheme. It is shown that the hiding capacity increases as \( p \) increases. Average hiding capacities of 24,212,667, 45,477.5, and 45,485.5 bits are achieved by the block-based histogram shifting scheme with \( 4 \times 4 \) blocks when \( p \) values are set to 1, 2, and 3, respectively. The gain of hiding capacity is quite small when \( p \) is increased from 2 to 3 or more. Therefore, it is suggested that the maximal \( p \) value should be set to 2.

The table of numbers of different pixels and different blocks in the tamper test.

<table>
<thead>
<tr>
<th>( p )</th>
<th>No. of different pixels</th>
<th>No. of different blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p=1 )</td>
<td>6146</td>
<td>456</td>
</tr>
<tr>
<td>( p=2 )</td>
<td>6148</td>
<td>456</td>
</tr>
<tr>
<td>( p=3 )</td>
<td>6148</td>
<td>456</td>
</tr>
<tr>
<td>( p=4 )</td>
<td>6148</td>
<td>456</td>
</tr>
</tbody>
</table>

3. Experimental results

Our experiments are performed on Windows 7 PC with an Intel Core i5 2.8 GHz CPU and the 2 GB RAM. The testing programs are implemented by using Bloodshed Dev C++. In our experiments, six grayscale images of \( 512 \times 512 \) pixels, “Airplane”, “Boat”, “Girl”, “Goldhill”, “Lenna”, and “Pepper” as shown in Fig. 4, are used. In the simulations, the block size is set to \( 4 \times 4 \).

Table 2 lists the hiding capacities of the proposed scheme. It is shown that the hiding capacity increases as \( p \) increases. Average hiding capacities of 24,212,667, 45,477.5, and 45,485.5 bits are achieved by the block-based histogram shifting scheme with \( 4 \times 4 \) blocks when \( p \) values are set to 1, 2, and 3, respectively. The gain of hiding capacity is quite small when \( p \) is increased from 2 to 3 or more. Therefore, it is suggested that the maximal \( p \) value should be set to 2.

The table of numbers of different pixels and different blocks in the tamper test.

<table>
<thead>
<tr>
<th>( p )</th>
<th>No. of different pixels</th>
<th>No. of different blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p=1 )</td>
<td>6146</td>
<td>456</td>
</tr>
<tr>
<td>( p=2 )</td>
<td>6148</td>
<td>456</td>
</tr>
<tr>
<td>( p=3 )</td>
<td>6148</td>
<td>456</td>
</tr>
<tr>
<td>( p=4 )</td>
<td>6148</td>
<td>456</td>
</tr>
</tbody>
</table>

In the tamper test, a butterfly as shown in Fig. 7(a) is added on the shoulder of each embedded image. The tampered object in the binary form is shown in Fig. 7(b). In the tamper
Fig. 10. Detected images of the proposed scheme. (a) Roughly detected image ($p=1$); (b) the refined image ($p=1$); (c) roughly detected image ($p=2$); (d) the refined image ($p=2$); (e) roughly detected image ($p=3$); (f) the refined image ($p=3$); (g) roughly detected image ($p=4$); and (h) the refined image ($p=4$).
test, there are 6217 pixels within the tampered object and 461 blocks of $4 \times 4$ pixels are affected.

These four tampered images when the $p$ values are set to 1–4 are shown in Fig. 8. The pixel difference images and the block difference images for the tamper test are listed in Fig. 9. From the results, some white spots within the tampered object are found in each pixel difference image. It indicates that some modified pixels within the butterfly have the same gray levels as the original pixels in the embedded image. The total numbers of different pixels and different blocks in the tamper test when the $p$ values are set to 1–4 are listed in Table 2. It is shown that 456 image blocks are tampered for these $p$ values.

Table 3

<table>
<thead>
<tr>
<th>$p$</th>
<th>Roughly detected image</th>
<th>Final refined image</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p=1$</td>
<td>208</td>
<td>382</td>
</tr>
<tr>
<td>$p=2$</td>
<td>286</td>
<td>409</td>
</tr>
<tr>
<td>$p=3$</td>
<td>286</td>
<td>409</td>
</tr>
<tr>
<td>$p=4$</td>
<td>287</td>
<td>409</td>
</tr>
</tbody>
</table>

The detected results of the proposed tamper detection procedure are listed in Fig. 10. The roughly detected images and the refined images are listed. No white spots are found within the modified areas in these final refined images. Compared to the tampered area as shown in Fig. 7(b), the tampered area of each refined image is clearly detected. However, some modified blocks in the boundary of the tampered area cannot be detected by using the proposed tamper detection procedure.

The total numbers of the tampered blocks in the roughly detected images and the final refined images by using the proposed scheme are listed in Table 3. The numbers of the tampered blocks in the roughly detected

Table 4

<table>
<thead>
<tr>
<th>Factors</th>
<th>$p=1$</th>
<th>$p=2$</th>
<th>$p=3$</th>
<th>$p=4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>TP</td>
<td>370</td>
<td>398</td>
<td>398</td>
<td>398</td>
</tr>
<tr>
<td>TN</td>
<td>91</td>
<td>63</td>
<td>63</td>
<td>63</td>
</tr>
<tr>
<td>FP</td>
<td>12</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>FN</td>
<td>15,911</td>
<td>15,912</td>
<td>15,912</td>
<td>15,912</td>
</tr>
</tbody>
</table>

Fig. 11. Results of the false detection for the tamper test. (a) False detection image ($p=1$); (b) false detection image ($p=2$); (c) false detection image ($p=3$); and (d) false detection image ($p=4$).
images are equal to 208, 286, 286, and 287 when the 
$p$ values are 1, 2, 3, and 4, respectively. There are 382, 409, 409, and 409 tampered blocks in the refined images when the 
$p$ values are 1, 2, 3, and 4, respectively.

Results of the false detected images of the proposed 
tamper detection procedure are listed in Fig. 11. The false 
detected image blocks are found in the boundary of the 
tampered area in each refined image. In addition, the 
analysis of the detecting accuracy for the tamper tests is 
listed in Table 4. The results of true positive (TP), true 
negative (TN), false positive (FP), and false negative are 
provided for these tests. There are 16,384 image blocks for 
each $512 \times 512$ image when the block size is set to $4 \times 4$. 
Recall that 456 blocks are affected when a butterfly is 
added into the embedded images in the tamper test. 
According to the results, 91, 63, 63, and 63 true negative 
blocks are detected for the first tamper test when the 
$p$ values are 1, 2, 3, and 4, respectively. In addition, 12, 11, 11, 
and 11 false positive blocks are detected for the first 
tamper test when the $p$ values are 1, 2, 3, and 4, 
respectively.

Results of the total numbers of the image blocks with 
different embeddable bits are listed in Table 5. The numbers of 
the blocks without any embeddable bits in the roughly 
detected images are equal to 5729, 3216, 3124, and 3123 
when the $p$ values are 1, 2, 3, and 4, respectively. There are 
34.967%, 19.629%, 19.067%, and 19.067% blocks of $4 \times 4$ 
pixels with no embeddable positions when the $p$ values are 1, 2, 3, 
and 4, respectively.

To understand the distribution of the embeddable loca-
tions in the proposed scheme, locations of the un-embeddable 
blocks in the test image “Lenna” are listed in Fig. 12. The

<table>
<thead>
<tr>
<th>$t$</th>
<th>$p=1$</th>
<th>$p=2$</th>
<th>$p=3$</th>
<th>$p=4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5729</td>
<td>3126</td>
<td>3124</td>
<td>3123</td>
</tr>
<tr>
<td>1</td>
<td>4219</td>
<td>3202</td>
<td>3202</td>
<td>3201</td>
</tr>
<tr>
<td>2</td>
<td>2800</td>
<td>2635</td>
<td>2637</td>
<td>2638</td>
</tr>
<tr>
<td>3</td>
<td>1723</td>
<td>2074</td>
<td>2073</td>
<td>2074</td>
</tr>
<tr>
<td>4</td>
<td>938</td>
<td>1566</td>
<td>1567</td>
<td>1567</td>
</tr>
<tr>
<td>5</td>
<td>511</td>
<td>1205</td>
<td>1205</td>
<td>1205</td>
</tr>
<tr>
<td>6</td>
<td>259</td>
<td>935</td>
<td>935</td>
<td>935</td>
</tr>
<tr>
<td>7</td>
<td>119</td>
<td>676</td>
<td>676</td>
<td>676</td>
</tr>
<tr>
<td>8</td>
<td>52</td>
<td>459</td>
<td>459</td>
<td>459</td>
</tr>
<tr>
<td>9</td>
<td>13</td>
<td>252</td>
<td>252</td>
<td>252</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>121</td>
<td>121</td>
<td>121</td>
</tr>
<tr>
<td>11</td>
<td>17</td>
<td>90</td>
<td>90</td>
<td>90</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>27</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>
number of un-embeddable blocks decreases as the $p$ value increases in the proposed scheme.

4. Conclusions

A novel reversible tamper detection scheme is proposed in the paper. A block-based authentication code generation procedure is designed in the proposed scheme because the hiding capacity of the reversible data hiding approach is limited. The authentication codes are then embedded into the residual values block by block by using the prediction-based histogram shifting process. Multiple pairs of peak and zero points can be selected to embed the authentication codes in the proposed scheme.

From the experimental results, good image qualities of the embedded images are obtained in the proposed scheme. Average embedded image qualities of 51.475 dB and 48.830 dB are achieved with the block size 4 × 4 when the $p$ values are set to 1 and 2, respectively. In addition, clear tampered shapes can be detected for the tampered images by using the proposed scheme. According to the false detected images of the proposed scheme, only few false detected blocks appear in the boundary of the tampered area.

Traditional irreversible image authentication schemes can only be employed to detect the tampered images of the general-purposed images. The proposed scheme provides the data reversibility and can be employed to protect the image integrity of the general-purposed images as well as the special-purposed images. In other words, the proposed image authentication scheme can be used to detect the tampered areas for the remote sensing images, the medical images, the military images, and so on.

Acknowledgments

This research was supported by Providence University, Taichung, Taiwan under contract the National Science Council, Taipei, R.O.C. under contract NSC 101-2221-E-126-014 and NSC 102-2221-E-126-010.

References


