

A High-Quality Reversible Image Authentication Scheme Based on Adaptive PEE for Digital Images

Thai-Son Nguyen^{1,2}, Chin-Chen Chang¹ and Tso-Hsien Shih³

¹Department of Information Engineering and Computer Science, Feng Chia University,
Taichung 40724, Taiwan, R.O.C,
[e-mail: thaison@tvu.edu.vn, alan3c@gmail.com]

²Department of Information Technology, Tra Vinh University
Tra Vinh Province, Vietnam

³Department of Computer Science and Information Engineering, National Chung Cheng University
Chiayi, Taiwan, R.O.C
[e-mail: jokergreem@gmail.com]

*Corresponding author: Chin-Chen Chang

*Received April 8, 2015; revised July 14, 2015; revised September 18, 2015; accepted October 14, 2015;
published January 31, 2016*

Abstract

Image authentication is a technique aiming at protecting the integrity of digital images. Reversible image authentication has attracted much attention of researcher because it allows to authenticate tampered regions in the image and to reconstruct the stego image to its original version losslessly. In this paper, we propose a new, reversible image authentication scheme based on adaptive prediction error expansion (PEE) technique. In the proposed scheme, each image block is classified into smooth or complex regions. Then, according to the characteristic of each block, the authentication code is embedded adaptively to achieve high performance of tamper detection. The experimental results demonstrated that the proposed scheme achieves good quality of stego images. In addition, the proposed scheme has ability to reconstruct the stego image to its original version, if no modification is performed on it. Also demonstrated in the experimental results, the proposed scheme provides higher accuracy of tamper detection than state-of-the-art schemes.

Keywords: Image authentication, tamper detection, fragile watermark, reversibility, high quality

1. Introduction

Since digital multimedia is transmitted and accessed in the Internet conveniently, digital images can be easily copied and illegally manipulated by digital image processing technologies. The authentication of the digital image is considered seriously, meaning that the modification on the content of the image is detected after arriving to the destination [1]. Recently, many watermarking techniques [1-15] have been proposed to authenticate the trustworthiness of digital content and to protect its integrity. By embedding the watermark, the image authentication can be applied to verify tampered regions in the image. Image authentication can be classified into three categories, i.e., multimedia hashing schemes, digital forensic schemes, and fragile watermarking schemes. In the multimedia hashing schemes [5-7], based on the content of the cover image, the hashed results is calculated and considered as the basic characteristic of the image. This hashed result is very distinctive to each image; therefore, it can be used for image authentication. However, to verify the integrity of the image by comparison, such schemes must append the hashed result with the original image before the image is transmitted to the receiver. In digital forensic schemes [8, 9], received multimedia can be verified without their original content by considering their intrinsic properties and traces to authenticate whether such multimedia are processed by any malicious operations. However, forensic schemes faced with the low accuracy and high computation complexity. Since fragile watermarking is very sensitive to modifications, therefore, it is suitable to apply for image authentication [10-18] by hiding a watermark into the image. In the receiver side, the watermark is extracted to prove the integrity of the received image. By using fragile watermarking, high accuracy of the tampered detection is achieved while maintaining good image quality of stego images. Many image authentication schemes based on fragile watermarking have been proposed in the last few decades. In 2001, Wong and Memon [10] proposed a fragile watermarking scheme based on secret and public keys for authenticating the tampered region and proving the ownership. In 2007, Zhang and Wang [11] embedded a set of tailor-made authentication code into the cover image and incorporated a statistical mechanism for locating the tampered pixels individually. In [12], a dual watermarking scheme is introduced for detecting tampered region in the image. This scheme can recover any regions that had been tampered in the image. To achieve the ability of recovering the tampered region, Chan [13] proposed a new image authentication algorithm based in hamming code. In this scheme, the parity check bits are generated by rearranging the bits of pixels in the image, which is used to reconstruct the value of the most-significant bit of each tampered pixel. However, this scheme still yielded low image quality of stego images. To further improve the image quality, Qin et al. [14] used image hashing algorithm and folding operation for image authentication. In this scheme, the restoration bits are generated by the adaptive bit allocation mechanism.

Besides above mentioned schemes [10-14], many state-of-the-art image authentication schemes based on fragile image watermarking techniques [15-19] have been proposed in the compression domain, i.e., vector quantization (VQ) and block truncation coding (BTC). Chuang and Hu [15] proposed an adaptive image authentication scheme for VQ-compressed images. In this scheme, the authentication code is randomly generated by a seed value and embedded into each VQ indices. To authenticate the given VQ compressed image, two sets of the authentication codes are required to perform the tamper detection operation. However, the image quality of stego image is reduced considerably. To achieve self-recovery of

VQ-compressed images, in 2013, Qin et al. [16] combined VQ algorithm and inpainting technique for image authentication and self-recovery. In Qin et al.'s scheme, each block is classified into a smooth block or a complex block. Then, VQ algorithm or inpainting technique is used for generating the recovery-bits. Instead of using VQ-compressed images for image authentication, Hu et al. proposed two new schemes [17, 18] by using BTC-compressed images. The scheme in [17], used permutation operation to embed the authentication code into bitmap several times, while the scheme in [18]

used a joint image compression for image authentication of BTC-compressed images. In the scheme [18], the authentication code is also embedded into the bitmaps of BTC-compressed image. Then, the BTC-compressed image is further compressed to decrease storage space significantly. However, the scheme offered the low image quality of stego images, when the average PNSR is less than 39 dB. To improve further quality of stego images and guarantee high accuracy of tamper-detection for BTC-compressed images, Nguyen et al [19] proposed a new image authentication scheme based on reference table. Each quantization level of the BTC-compressed image block is used for embedding the authentication code in their scheme. However, the image quality obtained by Nguyen et al.'s scheme is still limited, when the average PSNR is less than 45 dB.

It is observed that, in aforementioned state-of-the-art image authentication schemes, most of these schemes used irreversible data hiding techniques [20-22] for embedding the authentication code. Irreversible data hiding algorithms are used popularly for image authentication because of their superior properties, i.e., high embedding capacity, easy design and simple use, when compared with reversible data hiding algorithms [23-25, 28]. The main disadvantage of irreversible data hiding algorithms is that the cover image is distorted permanently. In other words, the cover image cannot be recovered to its original version after tamper detection. As a result, the schemes based on irreversible data hiding cannot be used in some special fields, i.e., fine artwork, military and medical images. This is because in such fields, the original version of cover images is very important and is required after tamper detection. Therefore, designing an image authentication with reversibility becomes a challenged issue. In [26], Lo and Hu first introduced an image authentication scheme based on reversible fragile watermarking scheme. In this scheme, histogram shifting (HS) technique is used for embedding authentication code. Since the scheme is based on HS technique, authentication code cannot be spread in the entire of the image. This is because some complex blocks are not embedded authentication code by HS technique, leading to the low accuracy of tampered detection. In addition, their scheme offered limited image quality.

It is motivated by the work done in [26], in this paper, we proposed a new, reversible scheme based on adaptive prediction error expansion for image authentication. In the proposed scheme, block classification is used to determine whether the current block is a smooth block or a complex block. Then, the prediction error expansion (PEE) [27] is used adaptively for embedding the authentication code into each block to achieve high accuracy of tamper detection and maintain good quality of stego images. The experimental results demonstrated that the proposed scheme achieves quite high accuracy of tamper detection, when the normalized correlation coefficient of stego images is always larger than 0.9. Meanwhile, the good image quality of the stego images is preserved.

The rest of this paper is organized as follows. Section 2 describes detail of the proposed scheme. Then, experimental results are discussed and analyzed in Section 3. Our conclusions are given in Section 4.

2. Proposed Scheme

In this section, we describe the proposed scheme in detail. First, a sequence of authentication code is generated by a seed K , and is embedded into the cover image to form the stego image. Then, the integrity of the stego image is authenticated by comparing the extracted authentication code with its original version. If some regions in the stego image are modified, meaning that the image is tampered, and the detected image is obtained to show where the image is tampered. Otherwise, if no regions are modified in the stego image, the cover image is reconstructed without any errors. Fig. 1 shows the flowchart of the proposed authentication code embedding phase.

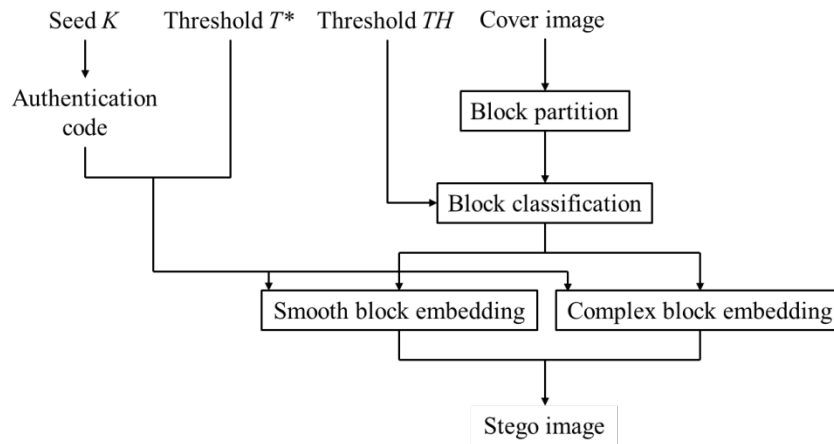


Fig. 1. Flowchart of the proposed authentication code embedding phase

2.1 Authentication code generation

Assume that the cover image I with the size of $W \times H$, which is first divided into non-overlapped blocks with the size of 3×3 . Therefore, there are totally $\lfloor W/3 \rfloor \times \lfloor H/3 \rfloor$ blocks, thus the number $\lfloor W/3 \rfloor \times \lfloor H/3 \rfloor$ of authentication codes will be constructed and embedded into the image. Here, we use a pseudo random number generator (PRNG) with the seed K to generate $\lfloor W/3 \rfloor \times \lfloor H/3 \rfloor$ random values. Then, each random value r is converted to binary information and embedded into each image block. Note that each block is classified into two different types, i.e., smooth and complex. Then, according to the type that the block belongs to, the adaptive way is used to embed the authentication code into the block. Details of block classification phase are described in the next subsection.

2.2 Block classification

For each block i with the size of 3×3 , Let C be a center pixel of the block, and its eight neighboring pixel be $E_1, E_2, E_3, E_4, T, B, L$, and R as shown in Fig. 2. The complexity of each block i is then calculated by using Equation (1).

$$Complexity_i = d_H + d_V, \quad (1)$$

where d_H and d_V are horizontal and vertical variances that are calculated by Equations (2) and (3).

$$d_H = |E_1 - E_2| + |E_4 - E_3|. \quad (2)$$

$$d_V = |E_1 - E_4| + |E_2 - E_3|. \quad (3)$$

According to the value of $Complexity_i$, the current block is classified into a smooth region or a complex region by comparing with a predefined complexity threshold TH . If $Complexity_i$ is greater than TH , the block is considered as the complex block. Otherwise, the block is considered as the smooth one.

E_1	T	E_2
L	C	R
E_4	B	E_3

Fig. 2. Illustration of the current block i

Fig. 3 shows the results of selected blocks on the image Barbara with different values of TH for embedding authentication code. The black and white regions in Fig. 3 correspond to the embeddable and un-embeddable blocks on the image Barbara, respectively. As can be seen from Fig. 3, the rich textures in the image are in the complex region and fewer bits of the authentication code are embedded. The larger the value of TH leads to fewer un-embeddable blocks and more embeddable blocks are used in this phase.

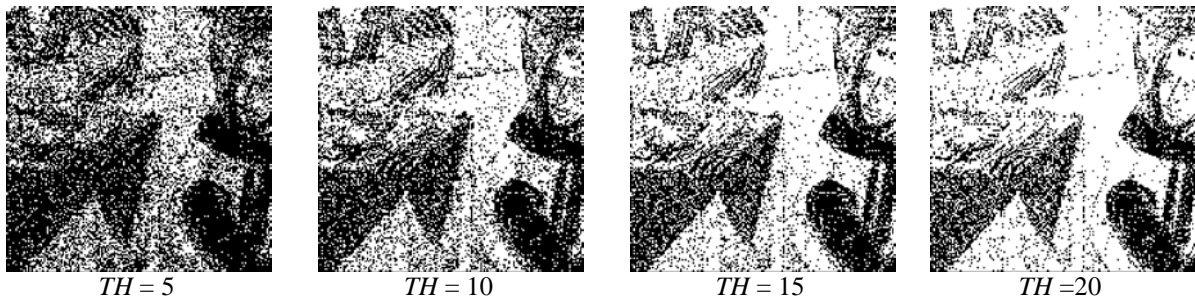


Fig. 3. Results of block classification on the image Barbara with different values of TH

2.3 Authentication code embedding

In this subsection, we describe how authentication code r is embedded into each block. As illustrated in Fig. 1, for a given cover image I with the size of $W \times H$ and three parameters, including the seed K , the embedding threshold T^* and the complexity threshold TH , the process of authentication code embedding is described as followings.

- Step 1: Divide the image I into non-overlapped blocks with the size of 3×3 , and generate $\lfloor W/3 \rfloor \times \lfloor H/3 \rfloor$ authentication codes by using Subsection 2.1 with the seed K .
- Step 2: For each image block i , according to the value of the complexity threshold TH , the block i is classified to the smooth block or the complex block as was done in Subsection 2.2.
- Step 3: The prediction errors d_T , d_B , d_L , and d_R of four neighboring pixels T , B , L and R of the center pixel C are calculated by Equation (4).

$$d_p = P - C, \quad (4)$$

where $P \in \{T, B, L, R\}$.

Step 4: To preserve the high quality of the stego images, we adopt two different embedding strategies to embed the authentication code r into the image block.

Step 4.1: For each smooth block, we extract four bits of the authentication code r by $w_1w_2w_3w_4 = \text{bin}(r \bmod 2^4)$, which are embedded into four prediction errors d_T , d_B , d_L , and d_R to generate four embedded prediction error d'_T , d'_B , d'_L , and d'_R by using Equation (5).

$$d'_P = \begin{cases} d_P \times 2 + w & \text{if } -T^* \leq d_P \leq T^* \\ d_P - T^* & \text{if } -T^* > d_P \\ d_P + T^* + 1 & \text{if } T^* < d_P \end{cases}, \quad (5)$$

where $w \in \{w_1, w_2, w_3, w_4\}$, $P \in \{T, B, L, R\}$, and T^* is a predefined embedding threshold. In the proposed scheme, to minimize the embedding distortion, we take threshold T^* as smallest one such that it is the ability to spread the authentication code over the entire of the image.

Step 4.2: For each complex block, only two bits w_1w_2 of authentication code r are determined and embedded into the image block. To avoid the significant distortion of the complex block when the authentication code is embedded, the values of horizontal and vertical variances, d_H and d_V , which are calculated in Equations (2) and (3), are used to determine the suitable pixels for embedding. If d_H is greater than d_V , then d_T and d_B are calculated by Equation (4) and they are embedded two bits w_1w_2 of the authentication code r by using Equation (5). Otherwise, d_L and d_R are used for carrying two bits w_1w_2 of the authentication code r .

Step 5: The value of four neighboring stego pixels, i.e., T' , B' , L' and R' , of the center pixel C are calculated as $P' = d'_P + C$.

Step 6: Steps 1 to 5 are implemented repeatedly, until the entire image blocks have been embedded authentication code completely.

We remark that, in the proposed scheme, the PEE technique is used for embedding the authentication code, thus, the overflow/underflow issues may be occurred during embedding process. Therefore, to prevent this shortcoming, the block is only used for embedding the bits of the authentication code, if the value of four neighboring pixels, i.e., T , B , L and R , of the center pixel C in the block is satisfied to the condition (6).

$$\begin{cases} 0 \leq P + 2 \times d_P + 1 \leq 255 & \text{if } -T^* \leq d_P \leq T^* \\ P < 255 - T^* & \text{if } d_P > T^* \\ P \geq T^* & \text{if } d_P < -T^* \end{cases}, \quad (6)$$

where P is the value of four neighboring pixels, i.e., T , B , L and R , of the center pixel C , and d_P is the corresponding prediction error. If Equation (6) does not hold, the block is not used for embedding the authentication code and its location is recorded in a location map, L . For example, for a given 512×512 host image, there are totally $\lfloor 512/3 \rfloor \times \lfloor 512/3 \rfloor = 28,900$ blocks that are needed to be embedded the authentication code bits. In the case of the current block cannot be hold the Equation (6), a bit 0 is required to indicate the current block. Otherwise, the bit 1 is used. Therefore, for difference types of image, i.e., the same size of the location map is obtained as 28,900 bits. Then, this location map is compressed by using JBIG-kit in [29]. Finally, the compressed location map L , two thresholds, TH and T^* , and the

seed K are also embedded into the image for reversibility. However, for security reason, the seed K is a secret key that is shared between the sender and the receiver in advance.

To avoid extra information in the proposed scheme, before embedding authentication code, the cover image I is partitioned into two different regions, i.e., the reversible region R_1 and the authentication region R_2 as shown in Fig. 4. The region R_1 is two first rows and two first columns of the image I that are used to embed the extra information, i.e., the compressed location map L , two thresholds TH and T^* , and the seed K . Whereas, the region R_2 is the remaining pixels that are used for carrying both the authentication code and the bit sequence S_{LSB} . By doing so, the bit sequence S_{LSB} should be merged into the authentication code as follows. Let $S_{LSB} = \{s_1, s_2, \dots, s_n\}$ be LSBs of pixels in the region R_1 , and the authentication code $C = \{c_1, c_2, \dots, c_{|C|}\}$. Then, we merge each bit of S_{LSB} and C together as $C^* = c_1||s_1||c_2||s_2||\dots||c_{|C|}$, that are embedded into the region R_2 , instead of only embedding the authentication code.

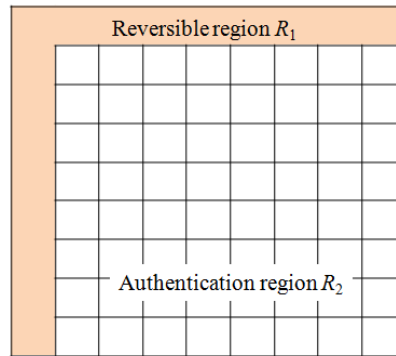


Fig. 4. Image partition

Note that the threshold T^* is essential to minimize the embedding distortion, it also plays important role in restricting the overflow/underflow issues in our scheme. By using adaptive PEE technique for embedding experimentally the authentication code into six different images, none of overflow/underflow problems is found in the proposed scheme, when TH is set from 5 to 20, and T^* is set from 0 to 4, respectively. Therefore, no location map is required. Accordingly, the size of extra information is only 24 bits, including two thresholds TH (8 bits) and T^* (8 bits), and the seed K (8 bits). In this scenario, using one least significant bit (LSB) of pixels in R_1 is enough to accommodate the extra information for reversibility.

2.4 Detecting and recovering

Once obtaining the stego image, to detect whether the image is tampered, we use the detecting and recovering algorithm. If the stego image is not modified, the image is recovered to its original version. Otherwise, the detected image will be generated to show where the image is modified. Fig. 5 shows the flowchart of the detecting and recovering phase.

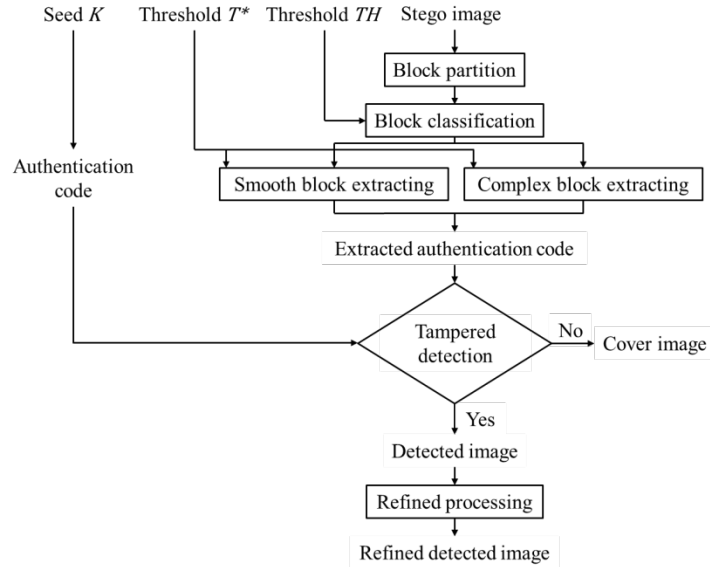


Fig. 5. Flowchart of the detecting and recovering phase

The detecting and recovering algorithm is described in detail as below.

Step 1: From the stego image, the location map L and the parameters, i.e., two thresholds, TH and T^* , and the seed K are extracted by reading LSBs of the region R_1 . Then, the authentication code C is generated PRNG with the seed K .

Step 2: For each image block i which is not recorded in the location map L , we calculate its complexity by using Equation (1). Then, the complexity value is compared with the threshold TH , to determine the block i is the smooth block or the complex block.

Step 3: The prediction errors d'_T , d'_B , d'_L , and d'_R of four neighboring stego pixels T , B , L and R of the center pixel C are calculated by Equation (4). According to the type of the block, i.e., smooth or complex, the embedded authentication bits are extracted as the following steps:

Step 3.1: For each smooth block, four original predicted errors d_T , d_B , d_L , and d_R are reconstructed by using Equation (7).

$$d_p = \begin{cases} \left\lfloor \frac{d'_p}{2} \right\rfloor, & -2T^* \leq d'_p \leq 2T^* + 1 \\ d'_p + T^*, & d'_p < -2T^* \\ d'_p - T^* - 1, & 2T^* + 1 < d'_p \end{cases}, \quad (7)$$

where $\lfloor \cdot \rfloor$ is a floor function. If $-2T^* \leq d'_p \leq 2T^* + 1$, one authentication bit w' will be extracted as $w' = d_p \bmod 2$. Otherwise, no authentication bit is extracted. Subsequently, four bits $w'_1 w'_2 w'_3 w'_4$ are extracted from d'_T , d'_B , d'_L , and d'_R , respectively.

Step 3.2: For each complex block, d_H and d_V are calculated and compared together. If d_H is greater than d_V , then d'_T and d'_B are used to extract two bits w'_1 , w'_2 of the authentication code. And the original values of d_T and d_B are determined by using Equation (8), while the original values of d_L and d_R are equal to d'_L and d'_R , respectively. Otherwise, two bits w'_1 , w'_2 of the authentication code are extracted from d'_L and d'_R . Equation (8) is used to calculate the values of d_L and d_R , whereas the original values of d_T and d_B are equal to d'_T and d'_B , respectively. In this case, two bits,

w'_3 and w'_4 , are set to 0. Therefore, irrespective of the smooth block or the complex block, four bits of the authentication code are extracted.

Step 4: After all groups of $w'_1w'_2w'_3w'_4$ of extracted authentication code C' are extracted completely from the stego image. Each four bits of the extracted authentication code C' are compared with the four corresponding bits $w_1w_2w_3w_4$ of the original authentication code C . If $w'_1w'_2w'_3w'_4 = w_1w_2w_3w_4$, the block is marked as a valid block, and the original value of four neighboring pixels, i.e., T , B , L and R , of the center pixel C are reconstructed as $P = d_p + C$, where $P \in \{T, B, L, R\}$. If $w'_1w'_2w'_3w'_4 \neq w_1w_2w_3w_4$, the block is marked as an invalid block.

Step 5: Repeat Steps 2 to 4 until the entire of blocks in the stego image has been detected completely. If no invalid block is encountered, the original image is reconstructed. Otherwise, we combine all of the valid blocks and the invalid blocks to generate the detected image.

Step 6: To further improve the accuracy of detection result, we employ a process of refinement; which will be executed several rounds. For each round, the process checks each valid block, marked as white color, in the detected image to convert it to the invalid block, marked as black color, according to four cases in Fig. 6. If one of four cases is matched, the color of the current block is set to black color. This process will be iteratively performed until there are not any blocks which turned to the invalid block.

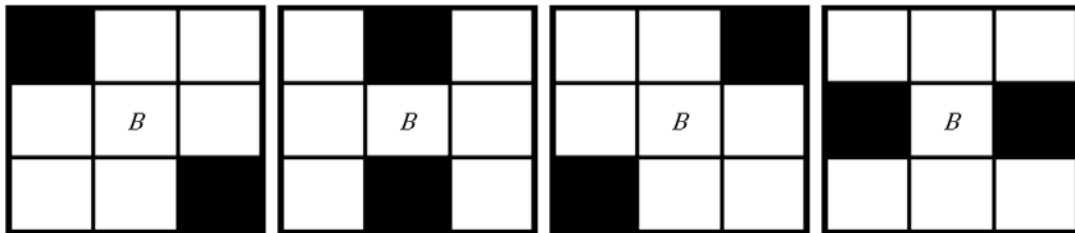


Fig. 6. Four cases of refinement process. (a) Case 1, (b) Case 2, (c) Case 3, (d) Case 4

3. Experimental Results

To demonstrate the performance of the proposed scheme, several experiments were performed on seven standard images with the size of 512×512 , including “Lena,” “Boat,” “Airplane,” “Toys,” “Goldhill,” “Barbara,” and “Baboon” (<http://sipi.usc.edu/database/>). Our programming was implemented on a PC with an Intel® Xeon® Processor E3-1230 v3 (8M Cache, 3.30 GHz), 8 GB of RAM with the operating system Windows 7 Ultimate 64-bit by Python 2.7.

Tables 1 and 2 show embedding capacity and image quality obtained by the proposed scheme, when different values of TH and T^* were used. Obviously, if the values of TH and T^* increased, the embedding capacity also increased in the proposed scheme, while the image quality decreased. The gain of embedding capacity was quite small while the image quality of the stego image decreased significantly, when the value of TH is increased from 10 to 15, Therefore, it is suggested that the threshold TH with value of 10 should be used in the proposed scheme to achieve high embedding capacity while maintaining good visual quality.

Table 1. Embedding capacity (bits) with different values of thresholds TH and T^*

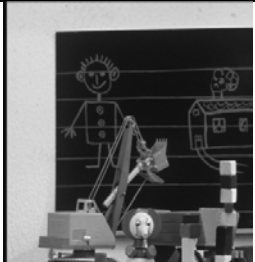
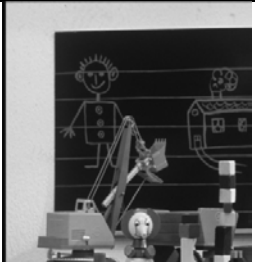
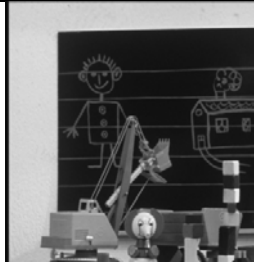
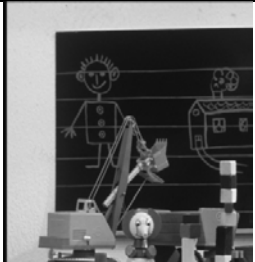
	Images	$T^*=0$	$T^*=1$	$T^*=2$	$T^*=3$	$T^*=4$
$TH = 5$	Lena	9,624	25,172	36,531	44,025	48,858
	Boat	7,404	21,171	32,151	40,174	45,707
	Airplane	13,578	33,573	44,569	51,039	55,054
	Toys	7,644	21,270	32,541	41,191	47,470
	Goldhill	6,835	18,884	28,478	35,591	40,986
	Barbara	4,852	14,070	21,910	28,242	33,096
	Baboon	4,035	7,952	14,697	17,121	20,926
	Average	7,710	20,299	30,125	36,769	41,728
$TH = 10$	Lena	12,083	31,961	46,539	56,045	61,978
	Boat	10,102	28,965	43,639	54,184	61,089
	Airplane	16,592	41,861	56,253	64,494	69,413
	Toys	10,033	27,967	42,993	54,276	62,389
	Goldhill	7,931	22,059	33,204	41,256	47,222
	Barbara	6,072	17,568	27,312	34,886	40,548
	Baboon	4,193	8,709	15,424	18,087	22,098
	Average	9,572	25,584	37,909	46,175	52,105
$TH = 15$	Lena	13,048	34,769	51,003	61,721	68,483
	Boat	11,036	31,501	47,442	58,890	66,370
	Airplane	17,399	44,195	59,879	69,104	74,685
	Toys	11,164	31,366	48,308	61,002	70,150
	Goldhill	8,489	23,637	35,657	44,415	50,921
	Barbara	6,710	19,321	30,062	38,393	44,596
	Baboon	4,361	8,981	16,167	19,075	23,273
	Average	10,315	27,681	41,217	50,371	56,925

Table 2. Image quality (dB) with different values of thresholds TH and T^*

	Images	$T^* = 0$	$T^* = 1$	$T^* = 2$	$T^* = 3$	$T^* = 4$
$TH = 5$	Lena	57.38	51.78	48.38	46.52	45.26
	Boat	57.25	51.64	47.85	45.83	44.45
	Airplane	57.00	51.76	48.54	46.75	45.51
	Toys	57.34	51.83	47.98	46.01	44.72
	Goldhill	57.53	51.77	48.01	45.90	44.45
	Barbara	57.55	51.76	47.58	45.26	43.62
	Baboon	57.68	51.04	47.19	44.65	42.79
	Average	57.39	51.65	47.93	45.84	44.40

$TH = 10$	Lena	56.49	50.62	47.58	45.78	44.60
	Boat	56.24	50.20	47.01	45.08	43.79
	Airplane	56.13	50.57	47.75	46.06	44.89
	Toys	56.30	50.22	47.00	45.09	43.86
	Goldhill	57.03	50.91	47.59	45.52	44.11
	Barbara	56.99	50.64	47.12	44.87	43.28
	Baboon	57.57	50.92	47.08	44.55	42.70
	Average	56.68	50.58	47.30	45.28	43.89
$TH = 15$	Lena	56.06	50.20	47.14	45.37	44.19
	Boat	55.91	49.88	46.69	44.77	43.50
	Airplane	55.82	50.23	47.39	45.69	44.54
	Toys	55.78	49.72	46.51	44.61	43.38
	Goldhill	56.72	50.59	47.27	45.21	43.80
	Barbara	56.69	50.35	46.86	44.64	43.07
	Baboon	57.44	50.80	46.97	44.45	42.60
	Average	56.35	50.25	46.98	44.96	43.58

Fig. 7 lists the quality of the different stego images with different values of TH and T^* . Here, the value of TH was set from 5 to 15, and the value of T^* was set 1 to 4, respectively. To evaluate the performance of the proposed scheme in terms of tamper detection, the stego image is modified by inserting the tampered object, a flower as shown in **Fig. 8(a)**, on the wall of each stego image. **Fig. 8(b)** shows the binary version of the tampered object.

Parameters	$T^* = 1$	$T^* = 2$	$T^* = 3$	$T^* = 4$
$TH = 5$				
	51.83 dB	47.98 dB	46.01 dB	44.72 dB









$TH = 10$				
	50.62 dB	47.58 dB	45.78 dB	44.60 dB
$TH = 15$				
	50.59 dB	47.27 dB	45.21 dB	43.80 dB

Fig. 7. Stego images and their quality with different values of TH and T^*

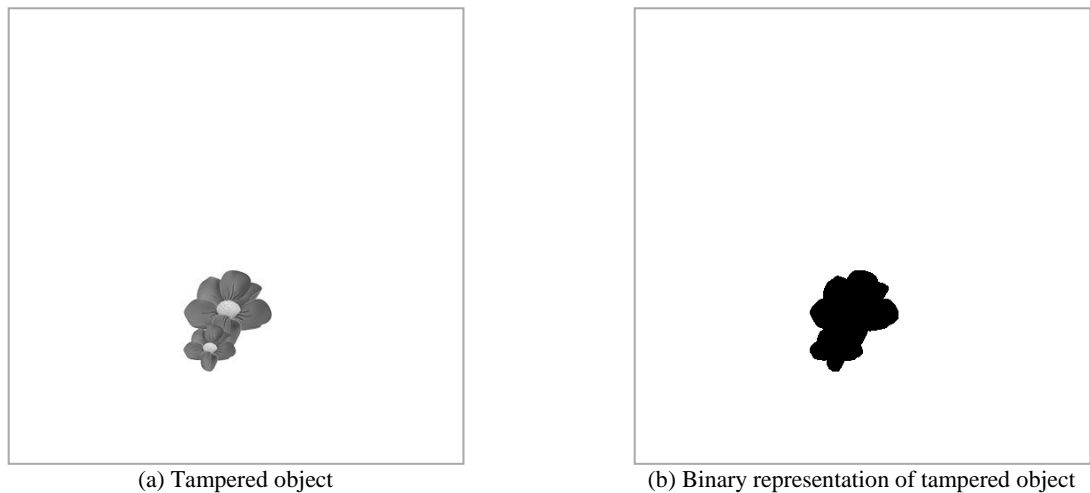


Fig. 8. Tampered object and its binary representation used in the tampered detection test in the proposed scheme

In the tamper detection test, three different images, i.e., Boat, Barbara, and Airplane, were used with different values of TH and T^* . For the images shown in **Fig. 8**, there are 7,346 pixels within the tampered object, and 886 blocks sized of 3×3 were modified in the stego images. **Fig. 9** shows the simulation results of the proposed scheme with different values of TH and T^* on different images when the tampered object has been inserted on the wall of each stego image. **Table 3** lists the total numbers of different pixels and different blocks in the tamper detection test when the different values of TH and T^* were used. As can be seen in this table, there are 838 blocks that are tampered in this test.

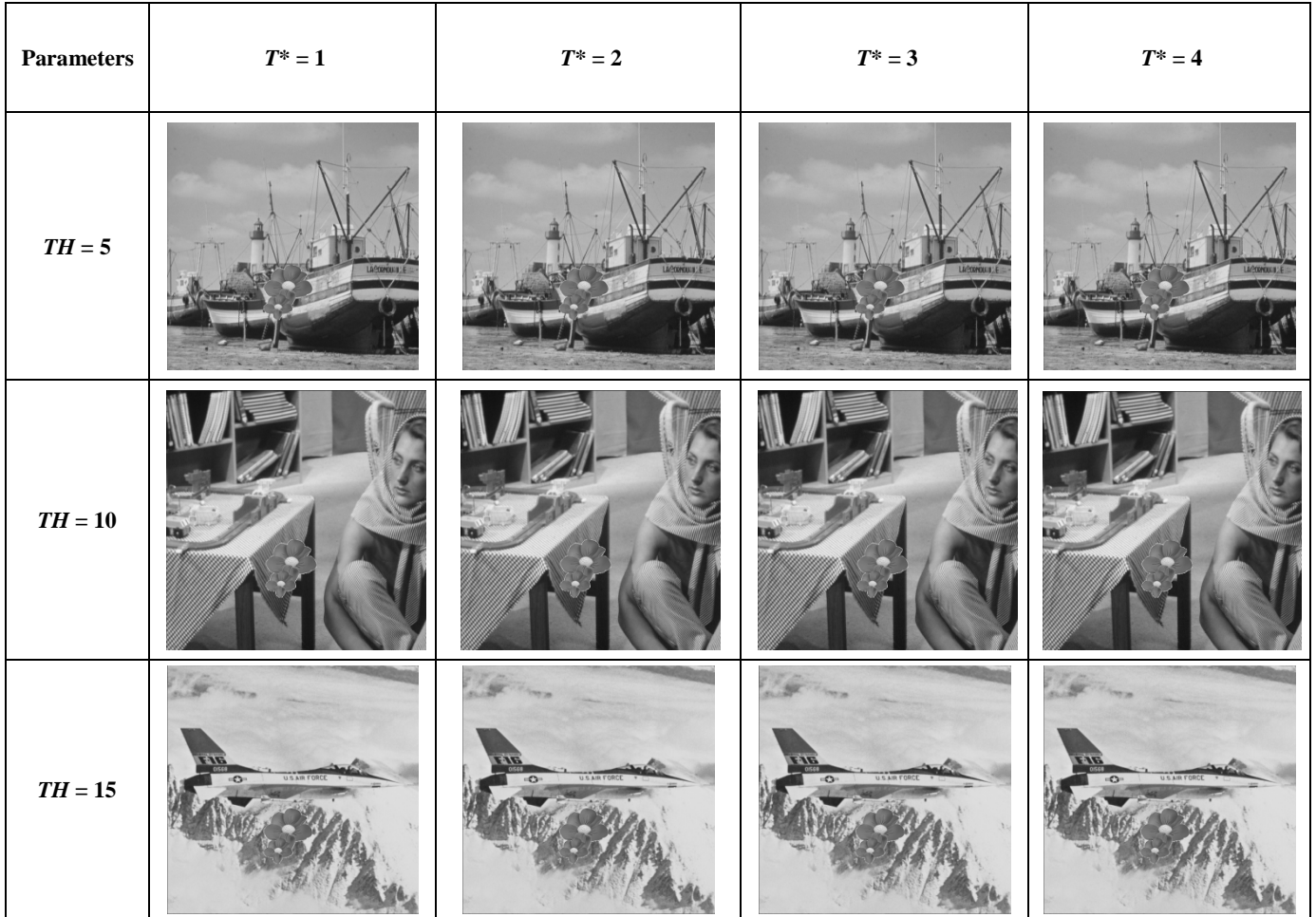


Fig. 9. Tampered images of the proposed scheme with different values of TH and T^*

Table 3. Numbers of different pixels and different blocks in the tampered detection test

TH	T^*	1	2	3	4
5	Number of different pixels	7,020	7,020	7,020	7,018
	Number of different blocks	838	838	838	838
10	Number of different pixels	6,985	6,983	6,983	6,980
	Number of different blocks	838	838	838	838
15	Number of different pixels	6,967	6,977	6,972	6,972
	Number of different blocks	838	838	838	838

Detected images and refined detected images obtained by the proposed scheme are shown in Fig. 10 and Fig. 11. As can be seen in Fig. 11, there is some white spots in the refined detected images when the value of $T^* = 1$ was used. However, no white spots were found in the refined detected image when the larger the value of T^* was used. In addition, when compared the results with the corresponding binary representation of tampered object, the tampered area of these refined detected images is completely verified. Table 4 presents the numbers of corresponding tampered blocks in the detected images and the refined detected images obtained by the proposed scheme. The number of tampered blocks in the detected images was significantly less than those of the refined detected images. This is because several white spots were occurred in the detected images, and these white spots were modified by using refinement process.













Parameters	$T^* = 1$	$T^* = 2$	$T^* = 3$	$T^* = 4$
$TH = 5$				
$TH = 10$				
$TH = 15$				

Fig. 10. Detected images of the proposed scheme with different values of TH and T^*

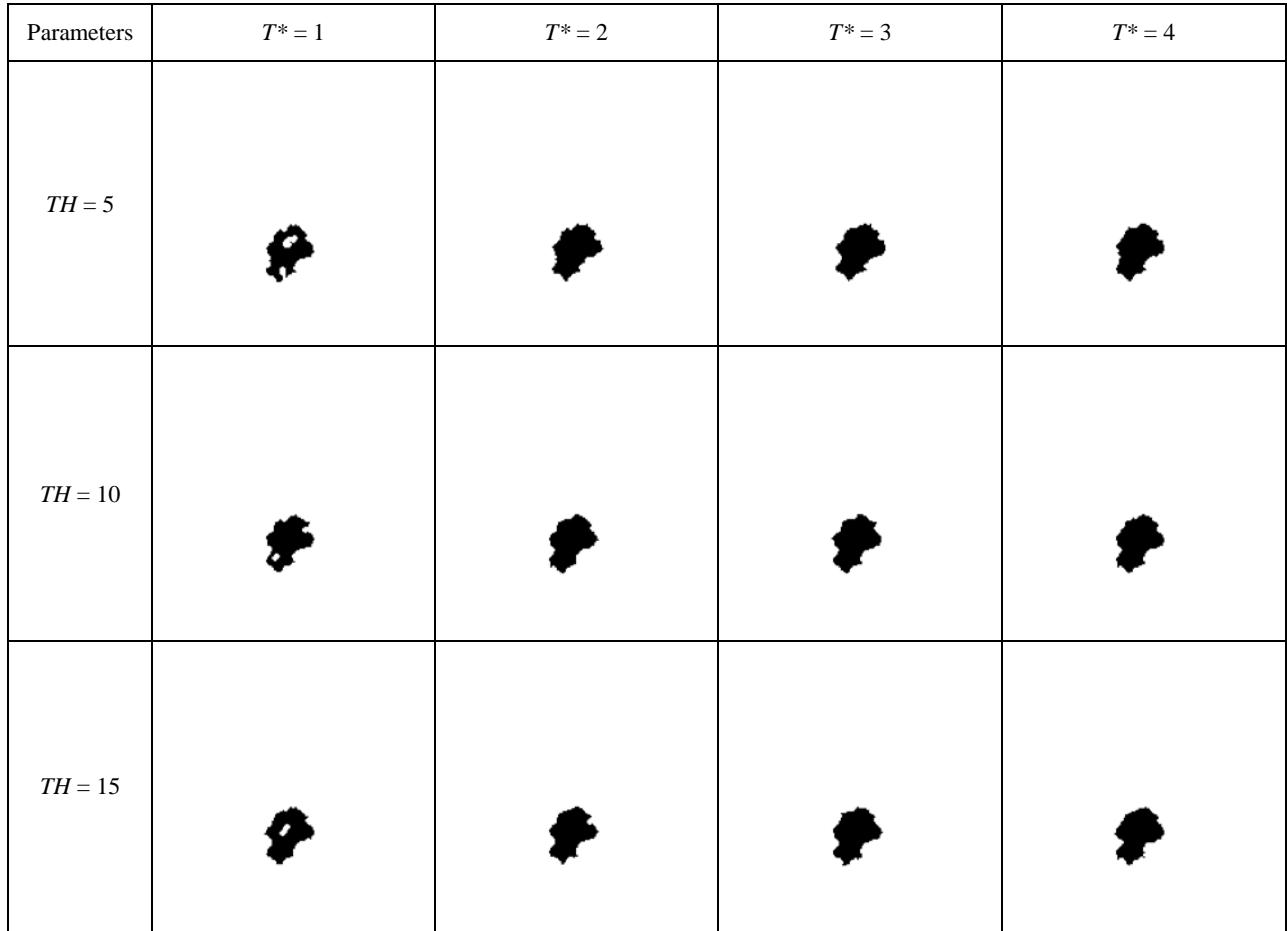


Fig. 11. Refined detected images of the proposed scheme with different values of TH and T^*

Table 4. Numbers of tampered blocks in detected images and refined detected images in the tampered detection test

TH	T^*	1	2	3	4
5	Detected image	361	433	466	495
	Refined detected image	621	713	723	735
10	Detected image	411	475	482	497
	Refined detected image	666	718	726	736
15	Detected image	416	473	492	507
	Refined detected image	670	711	730	743

To further estimate the results of the proposed scheme in the tamper detection test, we used the normalized correlation coefficient (NCC) to measure the similarity of the refined detected image and the binary representation of the tampered object. NCC is defined by Equation (9).

$$NCC = \frac{\sum_{i=1}^H \sum_{j=1}^W [B(i,j) - \bar{B}][RD(i,j) - \bar{RD}]}{\sqrt{(\sum_{i=1}^H \sum_{j=1}^W [B(i,j) - \bar{B}]^2)(\sum_{i=1}^H \sum_{j=1}^W [RD(i,j) - \bar{RD}]^2)}} \quad (9)$$

where B is the binary representation of the tampered image, RD is the detected image or the refined detected image, and H and W are dimensions of these images, respectively. The terms of \overline{B} and \overline{RD} denote the mean values of all pixels in B and RD , respectively. **Table 5** shows the NCC results of the proposed scheme according to the corresponding detected images and refined detected images shown in Figs. 10 and 11. As shown in this table, the high accuracy is obtained by the proposed scheme. Specifically, the NCC of the refined detected images in the proposed scheme is always larger than 0.9 in most cases.

Table 5. NCC results of the proposed scheme for the corresponding detected and refined detected images

TH	T*	1	2	3	4
5	Detected image	0.694	0.709	0.752	0.779
	Refined detected image	0.903	0.936	0.946	0.947
10	Detected image	0.711	0.766	0.787	0.783
	Refined detected image	0.909	0.944	0.949	0.950
15	Detected image	0.712	0.760	0.783	0.807
	Refined detected image	0.927	0.947	0.949	0.951

Our proposed scheme are also compared with some state-of-the-art schemes proposed by Hu et al. [18], Nguyen et al. [19], and Lo and Hu [26], as shown in **Table 6**. These three previous schemes were selected for comparisons because of their high performance in tamper detection. Here, $TH = 5$, and $T^* = 1$ are used in the proposed scheme. The table demonstrated the proposed scheme achieved higher average PSNRs of the stego images than other three schemes [18, 19, 26]. In addition, the proposed scheme has the ability to reconstruct the original image losslessly, if no tampered region was found in the stego image. Compared with Lo and Hu's scheme [26], although this scheme and the proposed scheme both obtained reversibility, but the proposed scheme provided the higher accuracy of tamper detection when the gain of average NCC was 0.027. The main reason is that Lo and Hu's scheme used the HS algorithm for embedding the authentication code. This leads to more authentication bits may be embedded in the smooth region, while fewer authentication bits, or even none of all, may be embedded into the complex region of the cover image. Conversely, according to block classification, the proposed scheme used adaptive embedding strategy for embedding the authentication code bits into the smooth and complex regions. By doing so, the authentication bits are spread over the entire of the cover image while preventing significant embedding distortion.

Table 6. Performance comparison of the proposed scheme with state-of-the-art schemes

Schemes	Block size	Average PSNR	Average NCC	Reversibility	The number of skipped blocks for embedding in average
Nguyen et al. [19]	4×4	40.58 dB	0.918	No	0
Hu et al. [18]	4×4	38.87 dB	0.903	No	0
Lo and Hu [26]	4×4	51.62 dB	0.911	Yes	6,493
Proposed	3×3	51.76 dB	0.938	Yes	4,925

To demonstrate the superiority of the proposed scheme, we compared the proposed scheme to the scheme developed by Sachnev et al. [28]. Fig. 12 shows the performance comparison of the proposed scheme and the previous scheme [28]. Referring to Fig. 12, the proposed scheme achieved better performance than the previous scheme [28] in each of the four images. The main reason is that the proposed scheme used the PEE technique adaptively for embedding secret bits into the smooth and complex regions of the host image, which guarantees that the value of pixels is modified as small as possible for embedding the secret data.

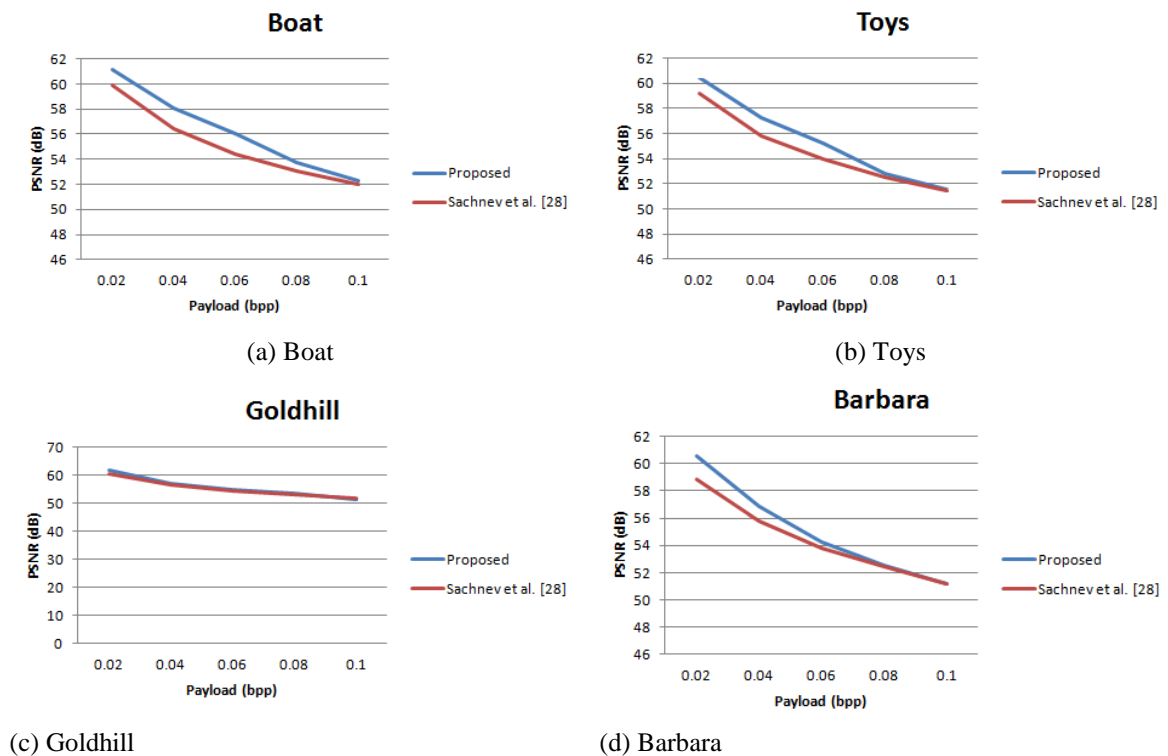


Fig. 12. Performance comparison of the proposed scheme and the previous scheme [28]

5. Conclusion

According to adaptive PEE technique, a reversible image authentication scheme for grayscale images is proposed. To guarantee that the authentication code spreading in the entire of the cover image, block classification is used to determine whether the current block is in a smooth region or in a complex region. Then, the PEE technique is applied adaptively for embedding the authentication bits to preserve high image quality. The experimental results demonstrated that the proposed scheme achieves good quality of stego images. In addition, the proposed scheme has ability to reconstruct the stego image to its original version, if no modification is performed on it. Also demonstrated in the experimental results, a clear tampered region (NCC larger than 0.9) is obtained in the proposed scheme, which outperformed three state-of-the-art schemes.

References

- [1] G. J. Yu, C. S. Lu, H. Y. M. Liao, "Mean quantization-based fragile watermarking for image authentication," *Opt. Eng.*, vol. 40, no. 7, pp. 1396–1408, 2001. [Article \(CrossRef Link\)](#)
- [2] H. T. Lu, R. M. Shen, F. L. Chung, "Fragile watermarking scheme for image authentication," *Electron. Lett.*, vol. 39, no. 12, pp. 898–900, 2003. [Article \(CrossRef Link\)](#)
- [3] C. T. Li, H. Si, "Wavelet-based fragile watermarking scheme for image authentication," *J. Electron. Imaging*, vol. 16, no. 1, 2007. [Article \(CrossRef Link\)](#)
- [4] H. J. He, J. S. Zhang, F. Chen, "Adjacent-block based statistical detection method for self-embedding watermarking techniques," *Signal Process.*, vol. 89, no. 8, pp. 1557–1566, 2009. [Article \(CrossRef Link\)](#)
- [5] A. Swaminathan, Y.N. Mao, M. Wu, "Robust and secure image hashing," *IEEE Trans. Infor. Forens. Secur.*, vol. 1, no. 2, pp. 215–230, 2006. [Article \(CrossRef Link\)](#)
- [6] V. Monga, M.K. Mhca, "Robust and secure image hashing via nonnegative matrix factorizations," *IEEE Trans. Infor. Forens. Secur.*, vol. 2, no. 3, pp. 376–390, 2007. [Article \(CrossRef Link\)](#)
- [7] D. Wu, X.B. Zhou, X.M. Niu, "A novel image hash algorithm resistant to print-scan," *Signal Process.*, vol. 89, no. 12, pp. 2415–2424, 2009. [Article \(CrossRef Link\)](#)
- [8] A. Swaminathan, M. Wu, K.J.R. Liu, "Digital image forensics via intrinsic fingerprints," *IEEE Trans. Infor. Forens. Secur.*, vol. 3, no. 1, pp. 101–117, 2008. [Article \(CrossRef Link\)](#)
- [9] H. Farid, "Exposing digital forgeries from JPEG ghosts," *IEEE Trans. Infor. Forens. Secur.*, vol. 4, no. 1, pp. 154–160, 2009. [Article \(CrossRef Link\)](#)
- [10] P. W. Wong, N. Memon, "Secret and public key image watermarking schemes for image authentication and ownership verification," *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1593–1601, 2001. [Article \(CrossRef Link\)](#)
- [11] X.P. Zhang, S.Z. Wang, "Statistical fragile watermarking capable of locating individual tampered pixels," *IEEE Signal Process. Lett.*, vol. 14, no. 10, pp. 727–730, 2007. [Article \(CrossRef Link\)](#)
- [12] T.Y. Lee, S.F.D. Lin, "Dual watermark for image tamper detection and recovery," *Pattern Recognition*, vol. 41, no. 11, pp. 3497–3506, 2008. [Article \(CrossRef Link\)](#)
- [13] C.S. Chan, "An image authentication method by applying Hamming code on rearranged bits," *Pattern Recognition Letters*, vol. 32, no. 14, pp. 1679–1690, 2011. [Article \(CrossRef Link\)](#)
- [14] C. Qin, C.C. Chang, P.Y. Chen, "Self-embedding fragile watermarking with restoration capability based on adaptive bit allocation mechanism," *Signal Processing*, vol. 92, no. 4, pp. 1137–1150, 2012. [Article \(CrossRef Link\)](#)
- [15] J.C. Chuang, Y.C. Hu, "An adaptive image authentication scheme for vector quantization compressed image," *J. Vis. Commun. Image Represent.*, vol. 22, no. 5, pp. 440–449, 2011. [Article \(CrossRef Link\)](#)
- [16] C. Qin, C.C. Chang, K.N. Chen, "Adaptive self-recovery for tampered images based on VQ indexing and inpainting," *Signal Processing*, vol. 93, pp. 933–946, 2013. [Article \(CrossRef Link\)](#)
- [17] Y.C. Hu, W.L. Chen, C.C. Lo, C.M. Wu, "A novel tamper detection scheme for BTC compressed images," *Opto-Electronics Review*, vol. 21, no. 1, pp. 137–146, 2013. [Article \(CrossRef Link\)](#)
- [18] Y.C. Hu, C.C. Lo, W.L. Chen, C.H. Wen, "Joint image coding and image authentication based on absolute moment block truncation coding," *Journal of Electronic Imaging*, vol. 22, no. 1, pp. 1–12, 2013. [Article \(CrossRef Link\)](#)
- [19] T.S. Nguyen, C.C. Chang, T.F. Chung, "A tamper-detection scheme for BTC-compressed images with high-quality images," *KSH Transactions on Internet and Information Systems*, vol. 8, no. 6, pp. 2005–2021, 2014. [Article \(CrossRef Link\)](#)
- [20] M. Iwata, K. Miyake, A. Shiozaki, "Digital steganography utilizing features of JPEG images," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E87-A, pp. 929–936, 2004. [Article \(CrossRef Link\)](#)
- [21] J. Mielikainen, "LSB matching revisited," *IEEE Signal Processing Letters*, vol. 13, pp. 285–287, 2006. [Article \(CrossRef Link\)](#)
- [22] W.J. Wang, C.T. Huang and S.J. Wang, "VQ applications in steganographic data hiding upon multimedia images," *IEEE Systems Journal*, vol. 5, no. 4, pp. 528–537, 2011.

[Article \(CrossRef Link\)](#)

- [23] J. Tian, "Reversible data hiding using difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 890-896, 2003. [Article \(CrossRef Link\)](#)
- [24] Z. Ni, Y. Q. Shi, N. Ansari, W. Su, "Reversible data hiding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, pp. 354-362, 2006. [Article \(CrossRef Link\)](#)
- [25] C. C. Chang, T. S. Nguyen, C. C. Lin, "Reversible image hiding for high image quality based on histogram shifting and local complexity," *International Journal of Network and Security*, vol. 16, no. 3, pp. 201-213, 2014. [Article \(CrossRef Link\)](#)
- [26] C.C. Lo, Y.C. Hu, "A novel reversible image authentication scheme for digital images," *Signal Processing*, vol. 98, pp. 174-185, 2014. [Article \(CrossRef Link\)](#)
- [27] D.M. Thodi and J.J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE Transactions on Image Processing*, vol. 16, pp. 721-730, 2007. [Article \(CrossRef Link\)](#)
- [28] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y.Q. Shi, "Reversible watermarking algorithm using sorting and prediction," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 19, no. 7, pp. 989-999, 2009. [Article \(CrossRef Link\)](#)
- [29] [Online]. Available: <http://www.cl.cam.ac.uk/~mgk25/jbigkit>. [Article \(CrossRef Link\)](#)



Thai-Son Nguyen received the bachelor degree in information technology from Open University, HCM city, Vietnam, in 2005. From December 2006, he has been lecturer of TraVinh University, TraVinh, Vietnam. In 2011, he received M.S. degree in computer sciences from FengChia University, TaiChung, Taiwan. He received the Ph.D. degree in computer science from Feng Chia University, Taichung, Taiwan. His current research interests include data hiding, image processing, and information security.



Chin-Chen Chang received the B.S. degree in applied mathematics and the M.S. degree in computer and decision sciences from National Tsing Hua University, Hsinchu, Taiwan, R.O.C., in 1977 and 1979, respectively. He received the Ph.D degree in computer engineering from National Chiao Tung University, Hsinchu, in 1982. From July 1998 to June 2000, he was Director of the Advisory Office, Ministry of Education, R.O.C. From 2002 to 2005, he was a Chair Professor at National Chung Cheng University. From February 2005, he has been a Chair Professor at Feng Chia University. In addition, he was served as a consultant to several research institutes and government departments. His current research interests include database design, computer cryptography, image compression, and data structures.



Tso-Hsien Shih, He is currently pursuing the M.S. degree with the Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan. His current research interests include data hiding, and image processing.