Indonesian Journal of Electrical Engineering and Computer Science
Vol. 21, No. 1, January 2021, pp. 253~262
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v21.i1.pp253-262

Reversible image authentication scheme based on prediction error expansion

Thai-Son Nguyen, Phuoc-Hung Vo
School of Engineering and Technology, Tra Vinh University, Tra Vinh, Vietnam

Article Info

Article history:

Received Mar 19, 2020
Revised Jul 21, 2020
Accepted Aug 21, 2020

Keywords:

Fragile watermark
Image authentication
PEE
Reversibility
Tamper detection

ABSTRACT

Reversible image authentication scheme is a technique that detects tampered areas in images and allows them to be reconstructed to their original version without any distortion. In this article, a new, reversible, image authentication scheme based on prediction error expansion is proposed for digital images. The proposed scheme classifies the host image into smooth blocks and complex blocks. Then, an authentication code that is created randomly with a seed is embedded adaptively into each image block. Experimental results showed that our proposed scheme achieves the high accuracy of tamper detection and preserved high image quality. Moreover, the proposed scheme achieved the reversibility, which is needed for some special applications, such as fine artwork, military images, and medical images.

Corresponding Author:

Thai-Son Nguyen
School of Engineering and Technology
Tra Vinh University
126 Nguyen Thien Thanh Str., Ward 5, Tra Vinh city, Vietnam
E-mail: thaison@tvu.edu.vn

1. INTRODUCTION

With the rapid Advancement of network and digital image processing technologies, this leads that copyright infringements can occur easily. For example, digital content can be copied illegally and modified maliciously when it is stored or transmitted over the Internet. Therefore, the protection of digital content has become an issue of increasing concern in both academia and industry [1, 2]. Recently, many authentication techniques [3-13] have been proposed to identify the trustworthiness of digital content and to protect its integrity. In principle, authentication techniques can be divided into two categories. The first category is hashing-based schemes [3-5] in which the hashed result of the image is calculated. Different images provided distinct hash results; therefore, hashing can be used for authentication. However, the hashed result must be appended with the original image, and sent to the receiver. To detect received images to be maliciously modified, the hashed result is re-calculated from the received image and compared with the appended hashed result. The second category is fragile watermarking schemes [6-19] that can obtain image authentication by embedding a watermark into the host image. Here, the watermark can be auxiliary information that can be generated by a pseudo random number generator (PRNG) with the seed. If the received image is suspected to have been tampered by malicious attackers, the watermark can be extracted to verify the tampered areas. In this category, the accuracy of the tampered areas and the visual quality of the stego image are two criteria of the fragile image watermarking techniques. The purpose of the earlier studies of fragile image watermarking techniques was to verify the integrity of the image's content in the spatial domain [7-9]. In 2011, Chan [7] introduced a new image authentication algorithm that used the hamming code to rearrange the bits of pixels.

If a pixel has been tampered, the most-significant bits of the pixel can be determined. Zhang et al. [8] proposed a novel watermarking scheme using the discrete cosine transform algorithm. If any modifications are found in the part of the watermarked image, the corresponding data for recovering the image are extracted from the area without any modification. In 2012, Qin et al. [9] proposed a new authentication technique to obtain high-quality restoration. Their scheme used image hashing algorithm for generating authenticationcode. Then, the adaptive bit allocation mechanism is used to encode the restoration bits. In addition, recent studies on fragile image watermarking techniques [10-13] have been leaded to the image in the compression domain, i.e., block truncation coding (BTC) and vector quantization (VQ).

From the literature, it is found that most of image authentication schemes are based on irreversible data hiding algorithms [20-22]. However, the distortion offered by irreversible data hiding is permanent, meaning that the embedded image cannot be reconstructed to its original version as reversible data hiding [23, 24]. To meet the requirement of being able to restore the host image completely after detection, Lo and Hu [25] proposed a reversible image authentication (RIA) scheme. Their scheme obtained the reversibility. However, the accuracy of the detection is inadequate high while the image quality is unsatisfactory. In [14], Yin et at. applied Hilbert curve for RIA scheme. Their scheme obtained higher detection rate and visual quality than those of Lo and Hu's scheme [25]. Later on, to maintain the integrity of digital images, Hong et al. [15] proposed new RIA scheme based on IPVO. In this scheme, according to the information of the unmodified pixels and the location, the hash code is is generated, and then embedded into modifiable pixels. By doing so, their scheme obtains the greater detection rate while ensuring the satisfactory visual quality of embedded images. Although the existing schemes effectively detect the tampered areas and can reconstruct the host image completely if they are un-tampered. However, these schemes fail to protect the modification in the complex blocks. To further improve the performance of existing schemes, in this article a new, RIA scheme is proposed for digital images. The PRNG with the seed is used for creating authentication code. To enhance the quality of the embedded image and to obtain more accurate detection, prediction error expansion (PEE) [26] is used adaptively for concealing the authentication code with smooth distribution characteristic. When there is reason to suspect that the image has been tampered by malicious attackers, the tampered areas are detected. If none of the blocks are modified, the original cover image is reconstructed exactly.

The remain of the paper is organized as follows. The proposd scheme is described in Section 2. Section 3 discusses the results and comparisons of the proposed scheme with the prior arts. Conclusions are presented in Section 4.

2. PROPOSED SCHEME

The main purpose of our proposed scheme is to detect whether an image has been tampered or not. If there are no modified or tampered areas in the image, the stego image is processed for reconstructing the original version of the host image. If some areas in the image have been modified, they will be detected. Figure 1(a) shows the framework of the proposed authentication scheme consisting of block classification, generation of the authentication code, and embedding of the authentication code.

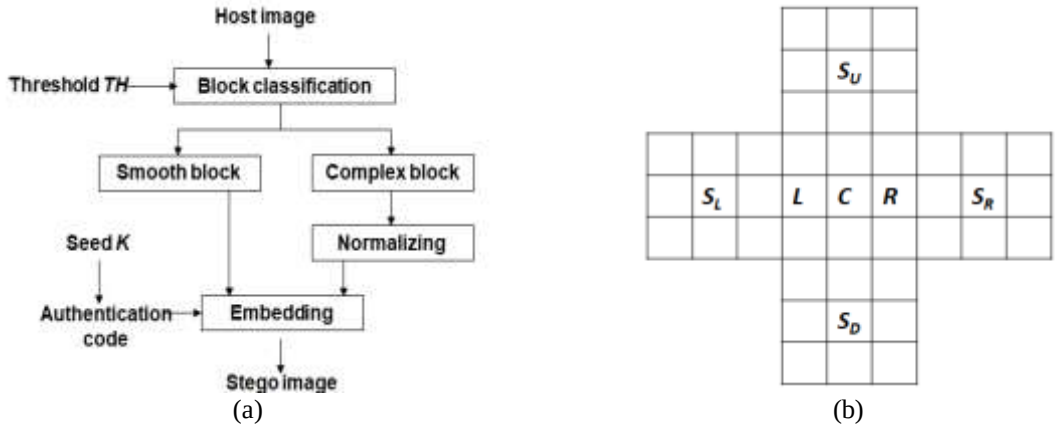


Figure 1. (a) Framework of the proposed authentication scheme; (b) Example of an image block and its satellite reference pixels

2.1. Block classification

Assume that a host image I of $W \times H$ pixels. Firstly, partition the host image into non-overlapping blocks of size 3 × 3. Let the center pixel C of the current block that is being processed be the reference pixel, and let L and R be the pixels that are located to the left and right of C, respectively. For non-border blocks, the reference pixel C has four satellite reference pixels, i.e., S_L, S_R, S_U, and S_D, which are located to the left, to the right, above, and below C, respectively, as shown in Figure 1(b).

$$Complexity = \max(|S_L - C|, |S_R - C|, |S_U - C|, |S_D - C|) \quad (1)$$

Higher complexity values are associated with blocks that are in areas with more complex textures. Note that according to (1), only the complexity of the non-border blocks can be calculated, thus, border blocks will not be processed in the proposed scheme. After the complexity values are obtained, they are compared with a classification threshold TH to determine whether the image block is in a smooth area or in a complex area as follows. Note that, for different type of block, the different way is used for embedding. Therefore, the threshold TH is also used to further improve the security of the proposed scheme.

a) Smooth area: $A_S = \{Bi \in I: Complexity\ (B_i) < TH\}$.
b) Complex area: $A_{Com} = \{Bi \in I: Complexity\ (B_i) \geq TH\}$.

2.2. Generation of the authentication code

With the host image sizes of $W \times H$ pixels, and block sizes of 3 × 3 pixels, a total of $k \times l$ image blocks will be obtained, where $k = W/3$ and $l = H/3$. Let two bits be embedded into each image block. Thus, to generate the authentication code sequence AC with size of $k \times l \times 2$ bits, the PRNG with a seed K is utilized to create $k \times l$ random values. Then, the random value rv_i is transformed to two bits, w_1w_2, of authentication code by using (2), and they are concatenated into the authentication code sequence AC.

$$w_1w_2 = bin(rv_i mod\ 2^2) \quad (2)$$

where $bin(\ )$ is the binary conversion function, w_1w_2 is two authentication bits that will be hidden into each image block i, and w_1w_2 is in {00, 01, 10, 11}. As we know that the reversible data embedding schemes provided much less embedding capacity than the irreversible data embedding schemes, thus, only two authentication bits are embedded into each image block.

2.3. Embedding the authentication code

After block classification and generation of the authentication, two bits w_1w_2 of authentication code sequence AC are embedded into each image block. The algorithm of the authentication code embedding is listed below:

a) Step 1: For each image block B_i, read two bits w_1w_2 from authentication code sequence AC.
b) Step 2: Compute the prediction errors d_L and d_R of the left and right adjacent pixels L and R of the center pixel C via $d_L = L - C$ and $d_R = R - C$, respectively.
c) Step 3: If $B_i \in A_S$, embed two bits w_1w_2 into the prediction errors d_L and d_R by using (3); here, bit w_1 is embedded into d_L and bit w_2 is embedded into d_R.

$$d' = \begin{cases} d \times 2 + w\ if\ -T^* \leq d \leq T^* \\ d - T^*\ if\ -T^* > d \\ d + T^* + 1\ if\ T^* < d \end{cases} \quad (3)$$

where d is the prediction error (d_L or d_R), and d' is the embedded prediction error (d_L' or d_R'), w is the authentication bit, i.e., w_1 or w_2, and T^* is the embedding threshold that is in the range 0 to 4.

d) Step 4: If $B_i \in A_{Com}$, compute the reference prediction error λ using (4). λ is used to normalize the current prediction errors d_L and d_R as small as possible, based on the satellite reference pixels, which guarantees that the proposed scheme can embed the authentication bits into the complex area without distorting the image significantly.

$$\lambda = min(|L^* - C|, |R^* - C|) \quad (4)$$

where $L^* = \left\lfloor \frac{C \times 2 + S_L}{3} \right\rfloor$ and $R^* = \left\lfloor \frac{C \times 2 + S_R}{3} \right\rfloor$ are calculated from the center pixel C and its two satellite pixels, S_L and S_R. Then, the current prediction errors d_L and d_R are normalized by:

$$d_L^* = d_L - \lambda \quad (5)$$

$$\text{and } d_R^* = d_R - \lambda \quad (6)$$

where d_L^* and d_R^* are two normalized prediction errors. Then, two bits w_1w_2 are embedded into d_L^* and d_R^* by (3) to generate the embedded prediction errors d_L' and d_R', respectively.

e) Step 5: Modify the pixel values of L and R to $L' = d_L' + C$ and $R' = d_R' + C$, respectively.

f) Step 6: Repeat Steps 1 through 5 until the image is processed completely.

In the proposed scheme, note that prediction error expansion (PEE) is used to embed the authentication bits.

To avoid overflow/underflow, only the pixels L and R of each block that satisfy the following conditions can be used to carry an authentication bit as shown in Figure 2.

$$\begin{cases} 0 \le C + 2d + 1 \le 255 \; if \; -T^* \le d \le T^* \\ C < 255 - T^* \; if \; d > T^* \\ C \ge T^* \; if \; d < -T^* \end{cases} \quad (7)$$

where C is the center pixel of the current block, T^* is the embedding threshold, and d is the corresponding prediction error of L or R. Otherwise, the pixels are skipped in the authentication code embedding process, and their block locations are recorded in a location map, LM. Then, the location map is processed to obtain reversibility. More discussion of the location map is presented in Subsection 2.5.

2.4. Tampered detection and restoration of the host image

Assume that the owner of the image suspects that a published image has been copied and modified from her/his image. In this scenario, such image is authenticated to verify whether to be modified or not. If the image has not been tampered, the original host image can be reconstructed completely after the authentication sequence is extracted. To extract and verify the authentication code, some system parameters, i.e., T^*, TH, and K, are required. Figure 2 shows a main steps of the tamper detection phase.

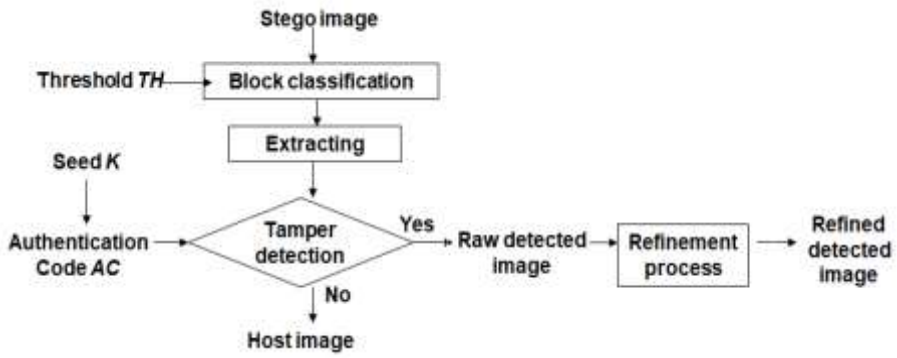


Figure 2. Main processes of tamper detection

Two authentication code sequences are generated for tampered detection. The first sequence AC is generated by using the PRNG with the seed K, as was done in Subsection 2.1. The second authentication sequence AC' is extracted from the embedded-image. After two authentication code sequences have been obtained, each two bits of AC and AC' are compared to determine whether the corresponding image block has been tampered or not. The tamper detection algorithm is shown in detail as follows:

a) Step 1: Generate AC by using PRNG with the seed K.

b) Step 2: For each block B_i, compute embedded prediction errors $d_L' = L' - C$ and $d_R' = R' - C$ of the left and right adjacent pixels L' and R' of the center pixel C, respectively.

c) Step 3: If $B_i \in A_S$, the original prediction errors d_L and d_R can be reconstructed as:

$$d_L = \begin{cases} \left\lfloor \frac{d_L'}{2} \right\rfloor \; -2T^* \le d_L' \le 2T^* + 1 \\ d_L' + T^* \; d_L' < -2T^* \\ d_L' - T^* - 1 \; d_L' > 2T^* + 1 \end{cases} \quad (8)$$

$$d_R = \begin{cases} \left\lfloor \frac{d'_R}{2} \right\rfloor & -2T^* \le d'_R \le 2T^* + 1 \\ d'_R + T^* & d'_R < -2T^* \\ d'_R - T^* - 1 & d'_R > 2T^* + 1 \end{cases} \quad (9)$$

where $\lfloor . \rfloor$ is the floor function. If d'_L and d'_R belong to $[-2T^*, 2T^* + 1]$, the authentication bits w_1' and w_2' can be extracted as $w_1' = d'_L mod\ 2$ and $w_2' = d'_R mod\ 2$, respectively.

d) Step 4: If $B_i \in A_{Com}$, the normalized prediction errors, d^*_L and d^*_R, can be calculated by:

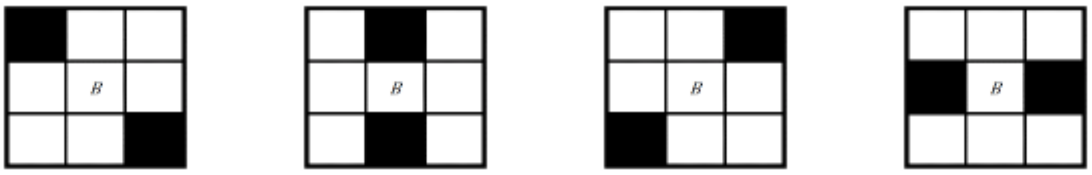
$$d^*_L = \begin{cases} \left\lfloor \frac{d'_L}{2} \right\rfloor & -2T^* \le d'_L \le 2T^* + 1 \\ d'_L + T^* & d'_L < -2T^* \\ d'_L - T^* - 1 & d'_L > 2T^* + 1 \end{cases} \quad (10)$$

$$d^*_R = \begin{cases} \left\lfloor \frac{d'_R}{2} \right\rfloor & -2T^* \le d'_R \le 2T^* + 1 \\ d'_R + T^* & d'_R < -2T^* \\ d'_R - T^* - 1 & d'_R > 2T^* + 1 \end{cases} \quad (11)$$

The authentication bits $w1'$ and $w2'$ also can be extracted as $w_1' = d'_L mod\ 2$ and $w_2' = d'_R mod\ 2$, respectively. Then, the extracted authentication code bits w_1w_2 are concatenated to the authentication code sequence AC'. Compute the reference prediction error λ using (4), as was done in authentication code embedding phase, and, then, the original prediction errors can be recovered as $d_L = d^*_L + \lambda$ and $d_R = d^*_R + \lambda$.

e) Step 5: Read two authentication bits w_1w_2 from the AC. If $w_1w_2 = w_1'w_2'$, the image block is marked as a clear block; otherwise, the image block is marked as a tampered block.
f) Step 6: Restore the original values of pixels L and R via $L = d_L + C$ and $R = d_R + C$, respectively.
g) Step 7: Repeat Steps 2 through 6 until all image blocks have been processed completely; then combine all the clear blocks and the tampered blocks to generate the raw detected image. If no tampered blocks are found, the host image is restored without any distortion.

It is clear that the above raw detected image should be further processed because, in the proposed scheme, some image blocks can not be used to contain authentication code bits because of the limited embedding capacity. Therefore, one refinement process should be used for the raw detected image. Each white block B is evaluated to be changed to a black block or not. To do so, the four test cases in Figure 3 were checked sequentially. For example, in the case 4 as shown in Figure 3(d), if the left and right adjacent blocks of B are black, then block B is colored black. Each white block in the raw detected image should be processed to construct the new refined detected image.



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

Figure 3. Four test cases for refinement process. (a) Case 1, (b) Case 2, (c) Case 3, (d) Case 4

2.5. Discussion of the location map

Figure 4 shows that the host image is divided into two regions, i.e., A_1 and A_2. The first region A_1 contains two first rows and two first columns of the image. This region is used to record the information of location map LM. The region A_2 consists of the rest of the pixels of the image which is embedded the authentication code bits and the LSB bits of the region A_1. Therefore, the LSBs of pixels in area A_1 must be extracted and merged into the authentication code sequence AC in advance. Let $\{a_1, a_2, \ldots, a_n\}$ be the set of LSBs of the region A_1 that are merged into the authentication code sequence $AC = \{w_1, w_2, \ldots, w_{|AC|}\}$ as $AC^* = w_1||a_1||w_2||a_2||\ldots||w_{|AC|}$. Then, instead of using authentication code sequence AC during the embedding phase, AC^* is used.

In the proposed scheme, avoiding the overflow/underflow problem is critical to the practical use of the proposed scheme; therefore, the location map is used. Table 1 shows the size of the location map that was

used during embedding authentication code with *TH* = 100 and various values of *T** into six grayscale images sized 512×512 [27], military images [28], and medical images [29]. In most cases, no location map is required. It can be seen that the largest size of the location map is required that is 144 bits for the Peppers image. However, more than 38,000 bits are embedded into this image as shown in Table 2, meaning that the embedding capacity is large enough to accommodate the location map. In addition, the location map is still compressed by using JBIG-kit in [30]. Then, the compressed location map *LM*, two thresholds, *TH* and *T**, and the seed *K* are also encrypted with the secret key *PK* and embedded into the image for reversibility. For security reason, the secret key *PK* is shared between the sender and the receiver in advance.

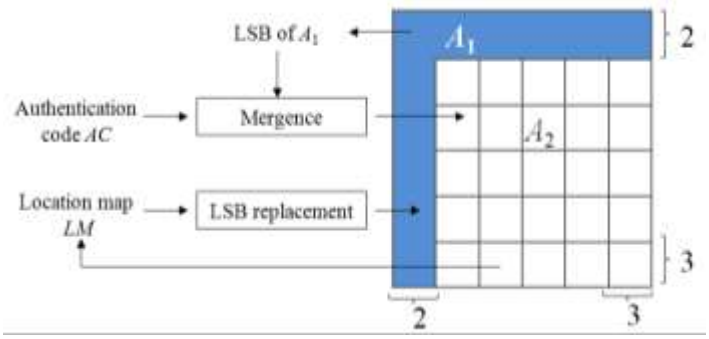


Figure 4. Image partition and location map embedding

Table 1. Size of the location map (bits) for various values of T*

T*	0	1	2	3	4
Image	Size of the location map (bits)				
Tank	0	0	54	108	144
Car and APCs	0	0	0	0	18
APC	0	0	0	0	54
MRI1	0	0	0	72	117
CT1	0	0	0	18	63
MRA1	0	0	0	0	27

3. RESULTS AND DISCUSSION

The proposed scheme was tested on publicly–available, standard images, including "Lena," "Boat," "Airplane," "Girl," "Goldhill," and "Peppers" [27]. Our computations were implemented on a PC with an Intel® Xeon® Processor E3-1230 v3 (8M Cache, 3.30 GHz), 8 GB of RAM. In the experiments, Windows 7 Ultimate 64-bit and by Python 2.7 are performed.

Table 2 shows the embedding capacity (EC) with various values of *TH* and *T**. It is clear that the EC of the proposed scheme increased when the thresholds *TH* and *T** increased. Average EC of 6,866; 18,801; 27,812; 34,244; and 38,837 bits were obtained for *TH* = 100 when *T** was set to 0, 1, 2, 3, and 4, respectively. The EC was slightly increased when the threshold *TH* was increased from 100 to 150. Figure 5 shows the visual quality of the stego images with various values of *T**, when *TH* = 100. The average visual quality of the embedded image decreased when the value of the threshold *T** increased. The PSNR of 51.72 dB and 49.90 dB was obtained with *T** = 0 and *T** = 1, respectively. Figures 6(a) and 6(d) show embedded images "Lena" obtained by the proposed scheme with *TH* = 100 and various values of *T**. In these four embedded images, the value of *T** was set from 0 to 3, respectively. In the tamper test, the tampered object in Figure 7(a) was inserted on the wall of each stego image, and its binary version is presented in Figure 7(b). Figure 8 shows that some white spots were found within the tampered object, meaning that some pixels in the tampered object had the same value as the original pixels in the stego images.

Table 2. Embedding capacity with various values of TH and T*

TH	*T** = 0	*T** = 1	*T** = 2	*T** = 3	*T** = 4
50	6,665	18,391	27,439	33,756	38,401
100	6,866	18,801	27,812	34,255	38,837
150	6,869	18,808	27,834	34,274	38,870

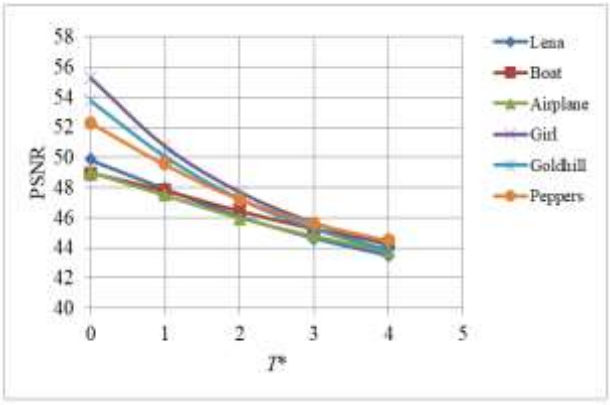
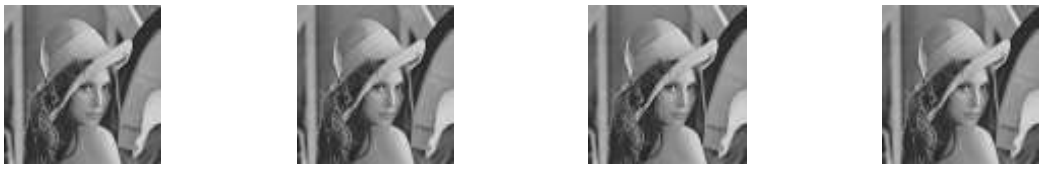


Figure 5. Image quality with difference values of T*



(a) PSNR = 51.41 dB, T* = 0 (b) PSNR = 48.94 dB, T* = 1 (c) PSNR = 46.27 dB, T* = 2 (d) PSNR = 44.64 dB, T* = 3

Figure 6. Embedded images (a-d) of the image "Lena" with various values of T*

(a) Tampered object 1 (b) Binary tampered object (c) Tampered object 2 (d) Tampered object 3

Figure 7. Tampered object used in the detection test

(a) Pixel difference image with T* = 0 (b) Block difference image with T* = 0 (c) Pixel difference image with T* = 1 (d) Block difference image with T* = 1

(e) Pixel difference image with T* = 2 (f) Block difference image with T* = 2 (g) Pixel difference image with T* = 3 (h) Block difference image with T* = 3

Figure 8. Difference images for tamper test

Figure 9 shows the detected results of the proposed schemes with various values of *T**. The left columns list the raw detected images, and the right columns list the refined detected images. No white spots were found in the refined detected images. In comparison with the binary version of the tampered object in Figure 7(b), the tampered region of each refined detected image is clearly determined, when the normalized correlation coefficient (*NC*) was always larger than 0.918 for different values of *T** as shown in Figure 9 while the average value of *NC* is 0.934 as shown in Table 3. *NC* can be calculated by (12).

$$NC = \frac{\sum_{i=1}^{H}\sum_{j=1}^{W}[TI(i,j)-TI_{mean}][DI(i,j)-DI_{mean}]}{\sqrt{(\sum_{i=1}^{H}\sum_{j=1}^{W}[TI(i,j)-TI_{mean}]^2)(\sum_{i=1}^{H}\sum_{j=1}^{W}[DI(i,j)-DI_{mean}]^2)}} \quad (12)$$

where *TI* is the tampered binary image, *DI* is the detected image, and *H* and *W* are the height and the width of the tamper binary image, respectively. The notations TI_{mean} and DI_{mean} are the average values of all pixels in *TI* and *DI*, respectively. In addition, to further estimate the accuracy of detection, we used F_1 score that is calculated using (13).

$$F\_1 = \frac{2\times Precision\times Recall}{Precision+Recall} \quad (13)$$

where *Precision* is the proportion of true positives among the sum of true positives and false positives and *Recall* is the proportion of true positives among the sum of true positives and false negatives, which are defined in (14) and (15), respectively.

$$Precision = \frac{True\ positive}{True\ positive + False\ positve} \quad (14)$$

$$Recall = \frac{True\ positive}{True\ positive + False\ negative} \quad (15)$$

As can be seen in Figure 9, the value of F_1 score obtained by the proposed scheme is greater than 0.915 for different values of *T**, meaning that the proposed scheme provided highly accurate tamper detection.

(a)Raw detected image with T* = 0, NC = 0.606, and F_1 = 0.547

(b) Refined detected image with T* = 0, NC = 0.919, and F_1 = 0.917

(c) Raw detected image with T* = 1,NC = 0.637, and F_1 = 0.586

(d) Refined detected image with T* = 1, NC = 0.922, and F_1 = 0.920

(e) Raw detected image with T* = 2, NC = 0.673, F_1 = 0.633

(f) Refined detected image with T* = 2, NC = 0.924, and F_1 = 0.923

(g) Raw detected image with T* = 3, NC = 0.695, and F_1 = 0.663

(h) Refined detected image with T* = 3, NC = 0.943, and F_1 = 0.942

Figure 9. Detected images of the proposed scheme

Figure 10 shows the test image "Lena" in the distribution of the embeddable (white color) and un-embeddable (black color) locations in the proposed scheme with *TH* = 100. Obviously, when *T** increases, the number of un-embeddable blocks decreases, meaning that more authentication code bits are embedded.

(a) T* = 0 (b) T* = 1 (c) T* = 2 (d) T* = 3

Figure 10. Distributions of embeddable and un-embeddable blocks in the image "Lena"

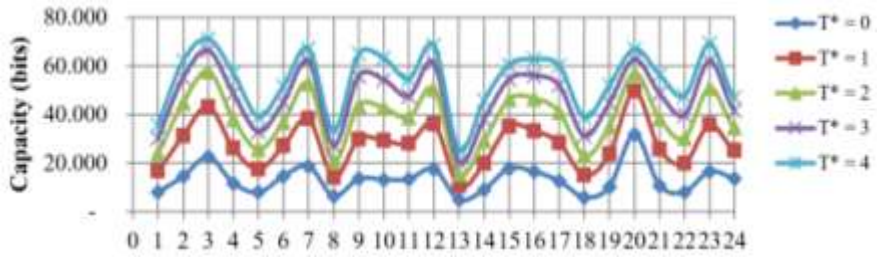
To justify the performance of the proposed scheme, five existing schemes [12, 14, 15, 22, 25] are compared with the proposed scheme in Table 3. In the tamper test, the tampered object in Figure 8(a) was inserted on the wall of twelve embedded images, i.e., six common test images [27], three military images [28], and three medical images [29]. Table 3 shows that the better PSNR value is obtained by our scheme among six schemes. In this paper, the average NC and F_1 score are used to estimate the detection accuracy. Moreover, to further evaluate the performance of the four schemes in detection accuracy, two tampered objects in Figure 7(c) and (d) are inserted in the wall of each image. As can be seen in Table 3, the higher detection accuracy is obtained by the proposed scheme, when the average NC and F_1 score both are greater than 0.910 when double tampered objects are used, while the those of other five schemes [12, 14, 15, 22, 25] are smaller than 0.905. In summary, the proposed scheme not only has several advantages over other five existing schemes but also offers high detection accuracy and comparable embedded image quality.

Figure 11 provides the EC and image quality of the grayscale versions of the 24 test images in the Kodak set (http://ww.r0k.us/graphics/kodak/), with *TH* = 50 and different values of *T**. As can seen in this

figure, the larger value of T^* is used, the higher EC is achieved and the more distortion is encountered. However, the average PSNR is larger than 39 dB when more than 54,000 bits has been embedded with $T^* = 4$.

Table 3. Comparison of the proposed scheme and the existing schemes [12, 14, 15, 22, 25]

Schemes	Block size (pixels)	Average PSNRs (dB)	Detection accuracy				Embedding technique	Reversibility
			Single tampered object		Double tampered object			
			Average *NC*	F_1 score	Average *NC*	F_1 score		
Hu et al. [12]	4 × 4	39.27	0.915	0.896	0.832	0.811	AMBTC modification	No
Nguyen et al. [22]	3 × 3	41.92	0.920	0.894	0.827	0.809	Reference table	No
Lo and Hu [25]	4 × 4	51.73	0.918	0.902	0.875	0.862	HS	Yes
Yin et al. [14]	4 × 4	51.82	0.921	0.912	0.889	0.887	IPVO	Yes
Hong et al. [15]	4 × 4	50.40	0.926	0.921	0.903	0.889	IPVO	Yes
Proposed	3 × 3	52.39	0.934	0.928	0.914	0.910	PEE	Yes



Embedding capacity of 24 Kodak images

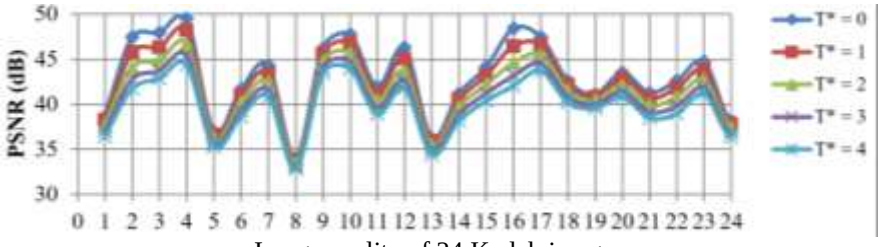


Image quality of 24 Kodak images

Figure 11. Performances of our scheme for 24 Kodak images with TH = 50 and different values of T*

4. CONCLUSION

In this article, a novel, RIA scheme is proposed by using PEE technique adaptively for embedding the authentication code. On the receiver side, the authentication code is extracted to detect tampered areas. If none of the blocks have been modified, the host image is reconstructed to its original version. Experimental results showed that the good image quality obtained by proposed scheme when the average PSNR of 52.39 dB and 48.90 dB when $TH = 100$ and $T^* = 0$ and $T^* = 1$, respectively. Moreover, the proposed scheme provided a clear tampered area and achieved reversibility. In addition, the proposed scheme achieved better results than other five existing schemes, in terms of the visual quality and the detection accuracy. Therefore, it should be suggested to be used for detecting tampered regions for special applications, i.e., fine artwork, military images, and medical images.

REFERENCES

[1] Luo, Z. Chen, M. Chen, X. Zeng, Z. Xiong, "Reversible image watermarking using interpolation technique," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 1, pp. 187-193, 2011.

[2] M. Boussif, N. Aloui, A. Cherif, "New Watermarking/Encryption Method for Medical Images Full Protection in m-Health," *International Journal of Electrical and Computer Engineering*, vol. 7, no. 6, pp. 3385-3394, 2017.

[3] S. Ababneh, R. Ansari, A. Khokhar, "Iterative compensation schemes for multimedia content authentication," *J. Vis. Commun. Image Represent.*, vol. 20, no. 5, pp. 303-311, 2009.
[4] M. G. Vargas, F. E. Hoyos, J. E. Candelo, "Portable and efficient fingerprint authentication system based on a microcontroller," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 4, pp. 2346- 2353 2019.
[5] H. Farid, "Exposing digital forgeries from JPEG ghosts," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 1, pp. 154-160, 2009.
[6] I. Q. Abduljaleel, A. H. Khaleel, "Hiding text in speech signal using K-means, LSB techniques and chaotic maps," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 6, pp. 5726 – 5735, 2020.
[7] 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.
[8] X. P. Zhang, Z. X. Qian, Y. L. Ren, G. R. Feng, "Watermarking with flexible self-recovery quality based on compressive sensing and compositive reconstruction*," IEEE Transactions on Information Forensics and Security*, vol. 6, no. 4, pp. 1223-1232, 2011.
[9] 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.
[10] 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.
[11] 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.
[12] 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.
[13] T. S. Nguyen, C. C. Chang, X. Q. Yang, "A reversible image authentication scheme based on fragile watermarking in discrete wavelet transform domain," *AEU-International Journal of Electronics and Communications*, vol. 70, no. 8, pp. 1055-1061, 2016.
[14] Z. Yin, X. Niu, Z. Zhou, J. Tang, B Luo, "Improved reversible image authentication scheme," *Cognitive Computation*, vol. 8, no. 5, pp. 890-899, 2016.
[15] W Hong, M.J. Chen, T. S. Chen, "An efficient reversible image authentication method using improved PVO and LSB substitution techniques," *Signal Processing: Image Communication*, vol. 58, pp. 111-122, 2017.
[16] D. C., Nguyen, T. S. Nguyen, F. R. Hsu, "An algorithm for DNA sequence hiding in H. 264/AVC video", *SoICT '16: Proceedings of the Seventh Symposium on Information and Communication Technology*, pp. 229-234, 2016.
[17] W. Hong, X. Y. Zhou, T. S. Chen, C. H. Hsieh, "An efficient reversible authentication scheme for demosaiced images with improved detectability," *Signal Processing: Image Communication*, vol. 80, 2020.
[18] Y. Y Peng, X. J. Niu, L. Fu, Z. X. Yin, "Image authentication scheme based on reversible fragile watermarking with two images," *Journal of Information Security and Applications*, vol. 40, pp. 236-246, 2018.
[19] G. Y. Gao, Y. Q. Shi, X. M. Sun, C. X. Zhou, Z. M. Cui, L. Xu, "Reversible Watermarking with Adaptive Embedding Threshold Matrix," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 9, pp. 4603-4624, 2016.
[20] J. Mielikainen, "LSB matching revisited," *IEEE Signal Processing Letters*, vol. 13, pp. 285–287, 2006.
[21] 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.
[22] T. S. Nguyen, C. C. Chang, and T. F. Chung, "A tamper-detection scheme for BTC-compressed images with high-quality images," *KSII Transactions on Internet and Information Systems*, vol. 8, no. 6, pp. 2005-2021, 2014.
[23] J. Tian, "Reversible data hiding using difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 890-896, 2003.
[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.
[25] C. C. Lo, Y. C. Hu, "A novel reversible image authentication scheme for digital images," *Signal Processing*, vol. 98, pp. 174-185, 2014.
[26] D. M. Thodi and J. J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE Transactions on Image Processing*, vol. 16, pp. 721-730, 2007.
[27] Miscellaneous Gray Level Images [Online]. Available (1/2015): http://decsai.ugr.es/cvg/dbimagenes/g512.php
[28] http://sipi.usc.edu/database/database.php. (Available on 07/10/2015)
[29] http:/www.osirix-viewer.com/datasets – DICOM sample image sets. (Available on 07/10/2015)
[30] [Online]. Available: http://www.cl.cam.ac.uk/~mgk25/jbigkit.