



# A Novel Semi-fragile Digital Watermarking Scheme for Scrambled Image Authentication and Restoration

Bin Feng<sup>1</sup> · Xiangli Li<sup>2,3</sup> · Yingmo Jie<sup>2,3</sup> · Cheng Guo<sup>2,3</sup>  · Huijuan Fu<sup>4,5</sup>

Published online: 3 January 2019  
© Springer Science+Business Media, LLC, part of Springer Nature 2019

## Abstract

More than ever, the practical and accurate watermarking technologies are called for the growing amount of exchanged digital image over the Internet. To protect the integrity and authenticity of digital image and to enhance the effect of tamper detection and restoration, we design and implement a semi-fragile watermark based on cat transformation, mostly used to locate tamper and recover for the transformed image and plain-image. The watermark which consists of two parts: the authentication watermark and recovery watermark, is embedded into the 2 least significant bit (LSB) of the pixel of the original image. The authentication watermark is calculated by the pixel value comparison and the parity check code, while the recovery watermark contains the average pixel value of the Torus image block. In the detection side, we use the hierarchy concept to locate the tamper in three layers and recover the attacked image in two layers. By using the hierarchy concept, this algorithm has another superiority that tamper can be detected on confused image. The experimental results show that our algorithm can accurately locate tamper and realize the content recovery and effectively prevent the vector quantization attack. Compared with other algorithms, this algorithm has better effect of tamper location and recovery.

**Keywords** Semi-fragile Watermark · Arnold scrambling · Hierarchical tamper location · Tamper recovery

## 1 Introduction

Facing the ever-growing quantity of digital images as an essential medium in the communications frequently and widely transmitted through the Internet, it becomes more and more critical to find the effective and practical technique of data hiding for intellectual property rights protections. Watermarking [1–5] is such a useful and pragmatic technique to prevent certain illegal actions, such as tampering, impermissible copying, or even unauthorized data integration. Applied to images, watermarking comes

down to embedding a hidden and invisible information as a watermark, which can be matched and retrieved in the process of detection even when it's tampered and attacked. As a scientific research for almost thirty years, it becomes the most common technique in image protection field. There are four principles driving the scientist in designing and implementing the watermarking algorithms: the invisibility, the robustness, the capacity and the security [24]. Any effective and practical algorithm should provide perfectly the enough balance among the four aspects. For instance, when the invisibility of the image is increased, the robustness of the watermarked image correspondingly decreases.

As an efficient tool of data hiding, the digital watermarking based on the above aspects relies on the fact that the human visual system (HVS) is not so sensitive to weeny change in the pixel values of the image [6–8]. Therefore, some useful information can be embedded into the original image by modifying the pixel values while humans can barely distinguish it. Additionally, the watermark can be applied to a lot of situation by using some specific algorithm [9].

Since the first watermarking algorithm was reported [1], there are numerous and different watermarking schemes derived and created. In brief, the algorithm [23] and schemes

---

A preliminary version of this paper appeared in Q-Shine 2017, 2017, December. This version includes a more detailed scheme and a concrete discussion, which was not included before.

---

✉ Cheng Guo  
guocheng@dlut.edu.cn

Bin Feng  
fengbin\_dl@163.com

Xiangli Li  
dllgdxlxl@foxmail.com

Extended author information available on the last page of the article.

depending on the domain in which the watermark is embedded, can be classified into the spatial domain [10–12], the transform domain [7, 8, 13–21], or a hybrid of those two domains [3, 22]. As it shows, in the spatial domain, the secret information of watermark is hidden in the pixels of the original image as a carrier. While in the transform domain, we can hide the information in the transform coefficient, such as Discrete Cosine Transform (DCT) or the Discrete Wavelet Transform (DWT). And for the hybrid algorithm, the information is embedded into the pixels or the coefficient, while there is some other information used to enhance the extraction process. While the image is transferred through the Internet and experience some helpful image processing between the embedding and extraction process, the quality of the digital image may be distorted and attacked, and the correctness of the image will be affected accordingly [23]. Thus, we should consider the robustness and the security of the image in the process of designing the watermark.

The attack in the image processing could be involved in several attacks proposed in watermarking techniques [25]. Sometimes, we need to authorize the image on the Internet and keep the security and confidentiality of the image simultaneously. Therefore, the security is focused in some research. Majority of digital image watermarking techniques use a secure key [26] to encrypt the order of the original image blocks or the watermark blocks; while valid users could use the secure key to extract the watermark or the original image, which usually are produced in the process of watermark embedding.

In this paper, we propose a new fragile watermarking algorithm based on the watermarking schemes [27, 28] by drawing on the experience of research methods [35–37]. By using a special hierarchical structure, the proposed method can resist the VQ codebook attack, while sustains the superior location properties and the public key structure of the original algorithm. In this scheme, the host image is divided into  $2 \times 2$  sized image blocks, which are the basic unit of generating authentication watermark and recovery watermark. As the above algorithm, we use a secure key to encrypt the order of image blocks and for an effective recovery, we modify the Torus method to map image blocks. After the embedding process, we change the cat transformation to suit for our algorithm and use the secure matrix as the key to encrypt the image which is embedded. This scheme is featured by authentication of the transformed image, and can keep the confidentiality of the image. The hierarchical model could be applied into the authentication process and recovery process. Besides, the proposed method based on [27] adds another part to recover from tampering by using the average pixel values of the image blocks. Through numerous experiments of this algorithm, our model has high robustness and security

compared with other methods. Besides, this algorithm also has the superior recovery effect, which is shown in the section of experimental results.

## 2 Overview of related work

Similarly to [30] and [31], we also use the Arnold transformation for different uses in the watermarking generation and scrambling transformation of image. In this chapter, we propose the related work of our method. We shall study the related work and change some parts of them to suit for our algorithm preferably.

### 2.1 Torus automorphism mapping

Torus isomorphic mapping is a typical chaotic map. In this method, a point is mapped to another different point, and for each point there is only one corresponding mapping point.

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = A \times \begin{bmatrix} x_n \\ y_n \end{bmatrix} \pmod{N} \quad (1)$$

$A$  is a matrix of  $2 \times 2$ , like  $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$  and  $\det A = 1$ .

In this paper, we use it for the selection of watermark embedded position. Instead of one point mapped to another point, we ameliorate the method for one image block mapped to another image block. Since the sequence of image blocks is a one-dimensional sequence, the Torus mapping is transformed into a one-dimensional transformation formula.

$$X' = f(X) = (k \times X) \pmod{N + 1} \quad (2)$$

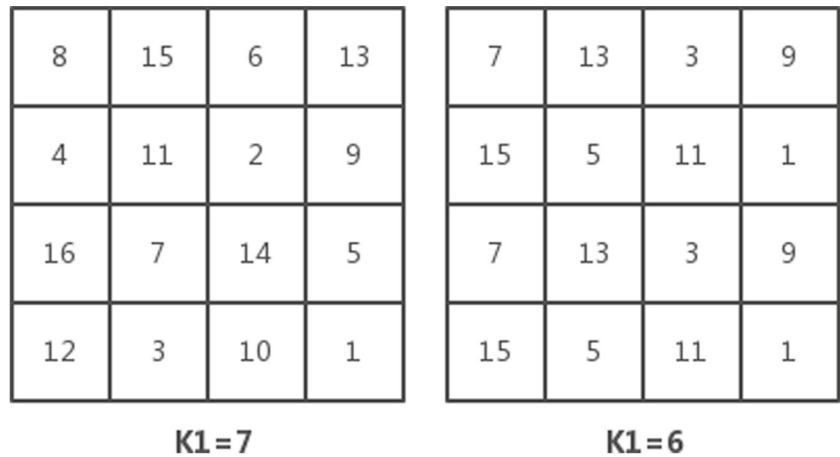
$X, X' (\in [1, N])$  are respectively the current serial number and the mapping number;  $k (\in [0, N - 1])$  must be a prime number and belong to a private key;  $N (\in \mathbb{Z} - \{0\})$  is the total number.

### 2.2 Arnold image scrambling algorithm

Arnold Scrambling is proposed by Russian mathematician Vladimir I. Arnold, also known as cat face transformation. Arnold scrambling has a periodicity, and after multiple transformations, the image will become very chaotic, but after specific transformations, the confused image will be transformed into the initial image. Such transformation can be used as image encryption [34].

In Arnold scrambling, the image is digitized into a matrix, and the rows and columns of its elements correspond to the values of the arguments, and the values of the

**Fig. 1** Torus mapping diagram ( $N = 16, k_1 = 7, k_1 = 6$ )



elements represent image information. The position  $(x', y')$  of the matrix in one transformation is

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \times \begin{bmatrix} x \\ y \end{bmatrix} \pmod{N} \tag{3}$$

$x, y \in \{0, 1, 2, \dots, N - 1\}$  indicates the position of the pixel before transformation. Digital images can be seen as a two-dimensional matrix, and after Arnold transformation, the pixel position will be rearranged, so the image will appear chaotic to achieve the effect of scrambling encryption.

### 3 Proposed method

In this paper, based on the literature [32, 33], our algorithm is proposed for hierarchical tamper location and restoration, which can be applied to both plain-image and scrambling images. Wherein tamper localization is based on the three-layer detection [32], and the effective recovery depends on the pixel information embedded in the Torus mapping block. The three-layer localization is carried out directly on the plain-image, and the tamper of the scrambling image can be detected on the cloud side, and we can decrypt the result and carry on the secondary detection for a better effect. The following sections describe the process of the watermark embedding, plain-image tamper detection, confused image tamper localization and recovery.

**Fig. 2** The 2 LSB of the pixels is set to zero

	8	7	6	5	4	3	2	1
Pixel 1							0	0
Pixel 2							0	0
Pixel 3							0	0
Pixel 4							0	0

### 3.1 Watermark embedding based on blocks

In this section, the original image is preprocessed to generate the watermark, and the watermark is embedded according to the Torus automorphism mapping. The watermark is embedded in the lowest 2 bits of each pixel.

#### 3.1.1 Pretreatment

Assuming the original image  $I$  is 256 gray levels, its size is  $M \times M$ , where  $M$  is a multiple of 2. The image is segmented and the block mapping sequence  $A \rightarrow B \rightarrow C \rightarrow D \rightarrow \dots \rightarrow A$  is obtained by the Torus automorphism transformation. Each letter in the sequence represents a separate block. That means the pixel value of block  $A$  is embedded in block  $B$ , the pixel value of block  $B$  is embedded in block  $C$ , and so on.

Firstly, we divide the image  $I$  into  $2 \times 2$  blocks and number them. Secondly, we use the random number functions to generate two prime key:  $k_1(k_1 \in [0, N]), k_2$ . Thirdly, we calculate their Torus mapping blocks by using  $k_1$  and the equation (2). In Fig. 1, if the key  $k_1$  is not a prime number, there may be more than one blocks mapped for one block. Besides, the second key  $k_2$  is used to be the security key for authentication watermark. Therefore, even though the attackers acquire the main function and the original image, the authentication watermark and recovery watermark are still unforgeable because of the confidentiality of  $k_1, k_2$ .

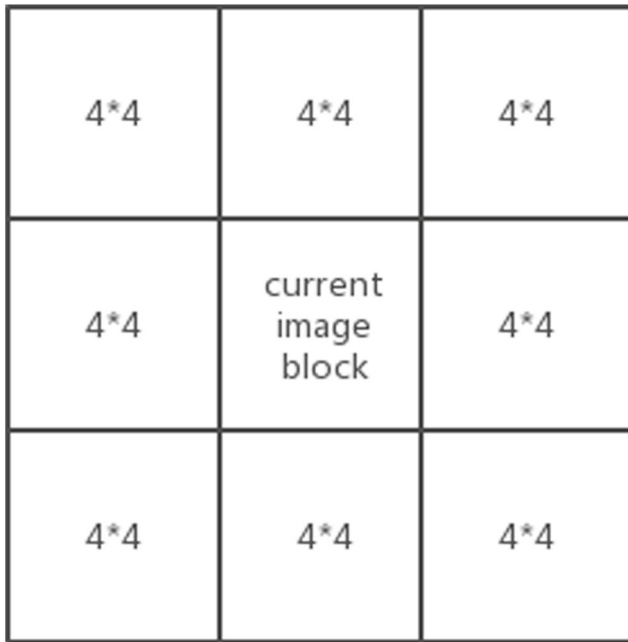


Fig. 3 Secondary tampering localization image block

### 3.1.2 Watermark generation and embedding

Assume  $A$  and  $B$  are a pair of Torus automorphism mapping blocks in the image  $I$ . And we have calculated the two secure keys  $k_1, k_2$ , where the Torus Mapping is based on the first secure key  $k_1$ .

The watermark of the image block  $B$  is represented by an array  $(v, p, r)$ , where  $v, p$  are one bit, and  $r$  is 6 bits

determined by the pixel value of  $A$ . The generation of watermark and the embedding process are as follows:

*Step 1 :* The 2 *LSB* of the pixels of  $B$  is set to zero like Fig. 2.

*Step 2 :* Generates authentication watermark  $v$  of the block  $B$ .

$$v = \begin{cases} 1 & B_{14} > B_{23} \\ 0 & B_{14} \leq B_{23} \end{cases} \quad (4)$$

*Step 3 :* Calculate the 6 bit *MSB* average  $B_{avg}$  of image block  $B$ .

*Step 4 :* Calculate the add code  $s$  by using the average  $B_{avg}$  and the second secure key  $k_2$ .

$$s = B_{avg} \oplus k_2. \quad (5)$$

*Step 5 :* Calculate the quantity  $N$  of 1 in  $s$ , and the parity watermark  $p$ .

$$p = \begin{cases} 1 & N \rightarrow \text{Even} \\ 0 & N \rightarrow \text{Odd} \end{cases} \quad (6)$$

*Step 6 :* The average  $A_{avg}$  of the 6 bit *MSB* of the image block  $A$  is as the recovery watermark  $r$ .

*Step 7 :* The watermark  $(v, p, r)$  are composed of 8 bits, and then embedded into the 8-bit *LSB* of the four pixels of the image block  $B$ .

Repeat the above steps (1) to (7) for the other blocks. Finally we can obtain the embedded image  $I'$  after watermark embedding.

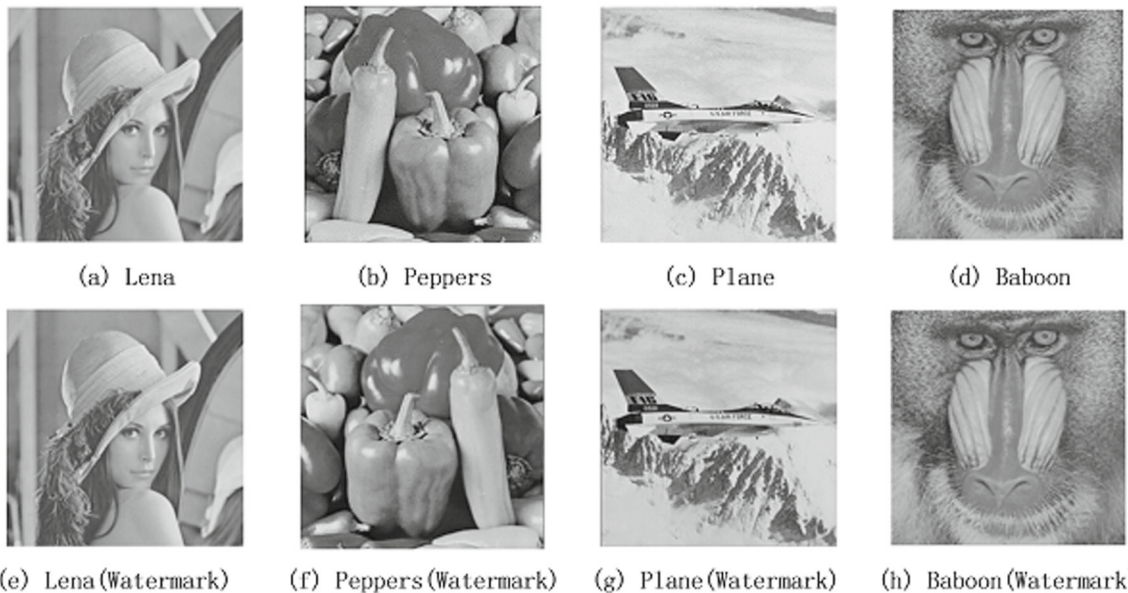


Fig. 4 The effect of the images embedded watermark

**Table 1** The *PSNR* and *SSIM* of images

Image	Lena	Peppers	Plane	Baboon
PSNR	47.16	47.10	47.29	47.53
SSIM	0.9795	0.9825	0.9777	0.9930

### 3.2 Arnold transformation

Firstly, we determine the transformation number of times  $N$  and the transformation matrix generator values  $a, b$ . And divide the embedded image  $I'$  into  $2 \times 2$  size image blocks. Take the image block  $A$  for an example.

**Step 1 :** The coordinate of the first pixel point of the block  $A$  are  $(x_A, y_A)$ , and the other coordinates are calculated as  $(x_A, y_A + 1)$ ,  $(x_A + 1, y_A)$ ,  $(x_A + 1, y_A + 1)$ .

**Step 2 :** By using the private key  $(a, b, N)$  and the follow equation, the coordinate  $(x_A, y_A)$  is converted to  $(x'_A, y'_A)$  after Arnold transformation.

$$\begin{bmatrix} x'_A \\ y'_A \end{bmatrix} = \begin{bmatrix} 1 & b \\ a & ab + 1 \end{bmatrix} \begin{bmatrix} x_A \\ y_A \end{bmatrix} \pmod{M} \quad (7)$$

**Step 3 :** The pixels  $(x_A, y_A + 1)$ ,  $(x_A + 1, y_A)$ ,  $(x_A + 1, y_A + 1)$  of the block  $A$  are respectively converted into  $(x'_A, y'_A + 1)$ ,  $(x'_A + 1, y'_A)$ ,  $(x'_A + 1, y'_A + 1)$ .

We repeat the above steps (1) to (3) for all image blocks for one transformation. Then, according to the number of times  $N$ , the Arnold scrambling image  $I'_{arnold}$  can be obtained by carrying on the operation for  $N$  times.

### 3.3 Tamper detection

#### 3.3.1 Tamper detection of plain-images

The tampered image  $I'_w$  is detected in three layers. In the first layer, we detect the  $2 \times 2$  image blocks. And in the second layer, we mark the independent  $4 \times 4$  blocks that has more than one marked  $2 \times 2$  block. In the third layer, mark the independent blocks according to the surrounding image blocks.

In the first detection, the image  $I'_w$  is divided into independent  $2 \times 2$  image blocks. Take the block  $B'$  as an example and the specific steps are as follows:

**Step 1 :** The watermark  $(v, p)$  in the image block  $B'$  is extracted according to the embedding rules.

**Step 2 :** Set the 2 bit *LSB* of the pixels of  $B'$  to 0, and calculate the average pixel value  $B'_{avg}$  of  $B'$ .

**Step 3 :** Calculate the code value  $s'$  according to the embedding progress with the secure key  $k_2$ .

**Step 4 :** Calculate the quantity  $N'$  of 1 in  $s'$  and the parity code  $p'$ .

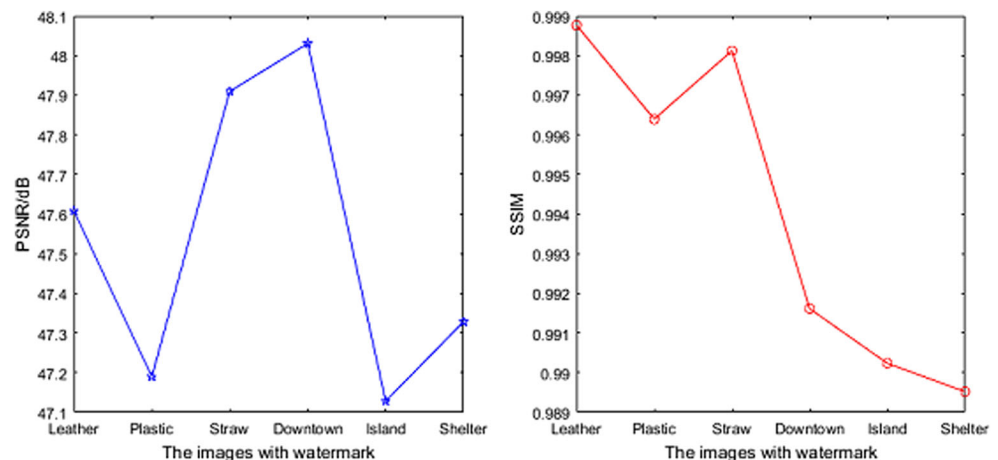
**Step 5 :** If  $p' = p$ , the image block  $B'$  is authenticated, otherwise the image block is marked.

**Step 6 :** When the parity code  $p'$  is verified, the image block  $B'$  is evaluated for the watermark  $v'$ .

**Step 7 :** If  $v' = v$ , the image block  $B'$  is authenticated, otherwise the image block  $B'$  is marked.

Repeat the above steps (1) to (7) for other image blocks of  $I'_w$ , and the detection result  $I_{locate}$  is acquired.

**Fig. 5** The *PSNR* and *SSIM* of the texture images and remote sensing images



In the second detection, the localization image  $I_{locate}$  is divided into independent  $4 \times 4$  image blocks and each individual image block is divided into four  $2 \times 2$  image blocks. And mark each individual  $4 \times 4$  image block that has more than one marked  $2 \times 2$  block, and finally obtain the second localization image  $I'_{locate}$ .

In the third detection, the second localization image  $I'_{locate}$  is divided into non-overlapping  $4 \times 4$  image blocks, and as shown in Fig. 3, the image block is marked where there are more than five marked image blocks of the eight surrounding blocks. After that, we get the final localization image  $I''_{locate}$ .

### 3.3.2 Tamper detection of scrambled images

Assume the the scrambled image  $I'_{Arnold}$  requires tamper detection in an unsafe third party, the insecure cloud detection system is  $A_{cloud}$ , and the local security detection system is  $B_{locate}$ . The confused image is detected in the first layer in the cloud detection system.  $A_{cloud}$  send detection results to the local security detection system for 2,3 layer detection. The specific steps are as follows:

Step 1 : At the cloud system, calculate the location image  $I'_{locate}$  like section 2.3.1.

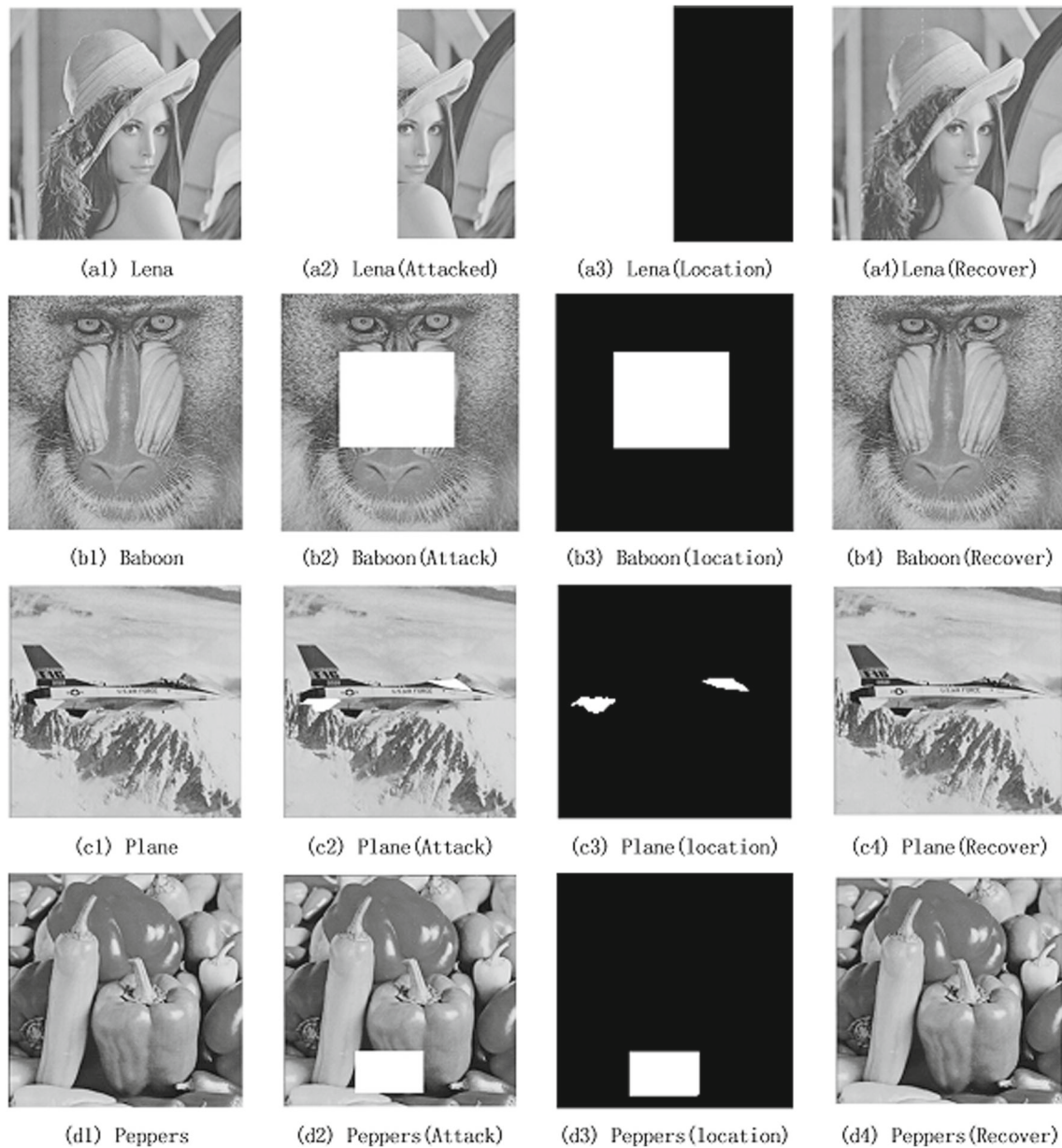


Fig. 6 The effect of location and recovery

**Table 2** The *PSNR* and *SSIM* of images

Image	Lena	Baboon	Plane	Peppers
PSNR	35.5592	39.1158	49.8609	45.7648
SSIM	0.8925	0.9563	0.9981	0.9879

**Step 2 :** In the local detection system, use the private key to decrypt  $I_{locate}^{cloud}$  to get the location image  $I_{locate}^{location}$ .

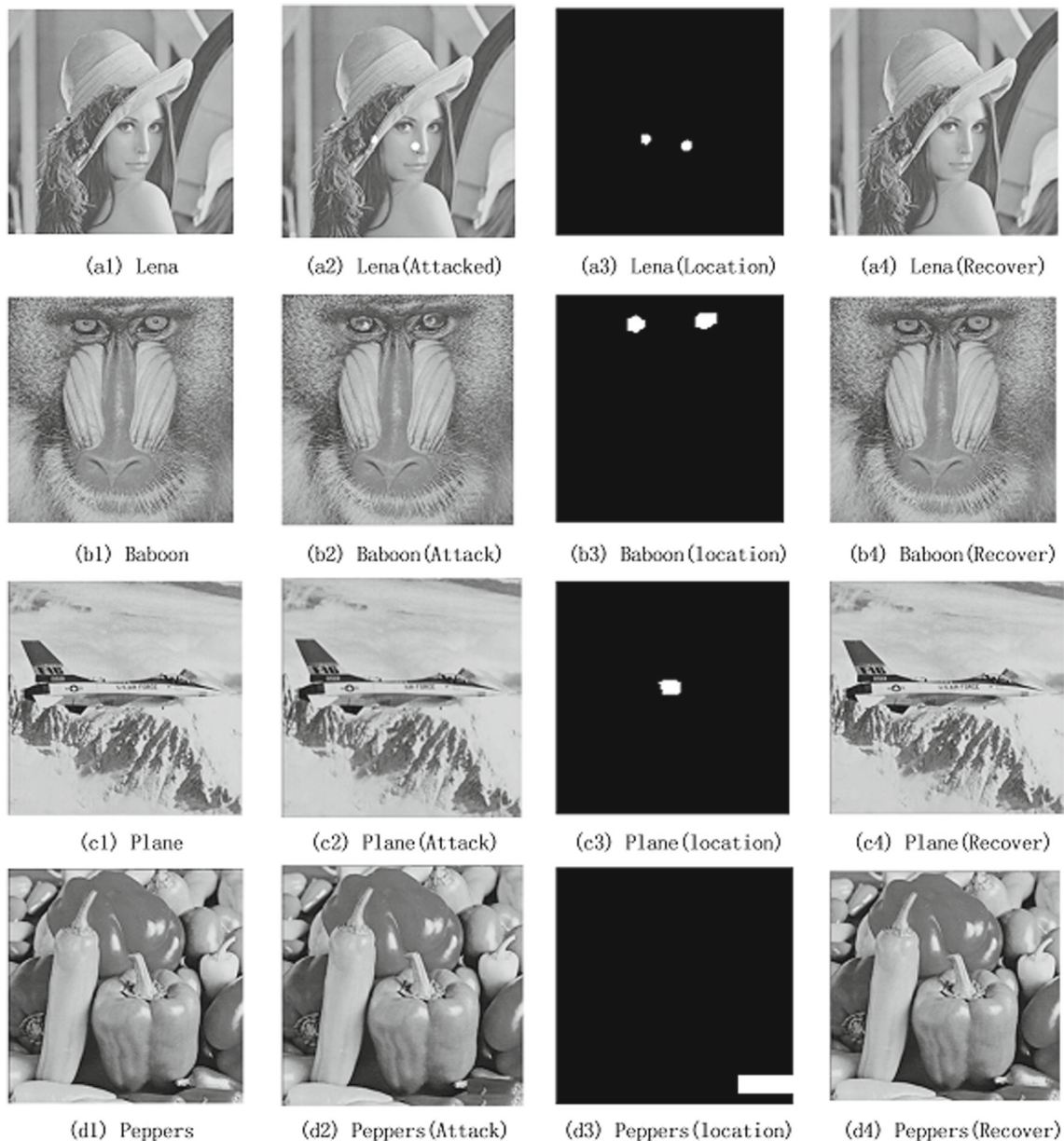
**Step 3 :** In the second detection, the location image  $I_{locate}^{location}$  is divided into independent  $4 \times 4$  image blocks and it is detected whether there is a marked independent  $2 \times 2$  image block in each

individual  $4 \times 4$  image block. And finally get the second location image  $I_{locate}^{location}$ .

**Step 4 :** In the third detection, the location image  $I_{locate}^{location}$  is divided into  $4 \times 4$  image blocks. Mark the image block where there are more than five marked surrounding image blocks, and finally get the location image  $I_{locate}^{location}$ .

### 3.4 Tamper recovery

After the above tamper detection, we need to recover the image. Firstly, divide the image  $I_{attack}$  into independent  $2 \times 2$  size image blocks. And for every tampered image block, we

**Fig. 7** The effect of location and recovery

carry on the recovery operations. Assume the image block  $B'$  has a tamper mark, and take  $B'$  for an example.

- Step 1* : The image block  $C'$  is calculated according to the key  $k$  of the Torus transformation.
- Step 2* : If the image block  $C'$  is not marked with tamper, extract the recovery watermark  $r$ , shift  $r$  left twice, and get  $r'$  to recover the image block  $B'$ .
- Step 3* : The pixel value of the image block  $B'$  is replaced with  $r'$ .
- Step 4* : If the image block  $C'$  has a tamper mark, the image block  $B'$  is re-marked.

Repeat the steps (1) to (4) for all the image blocks, and finally obtain the recovery image  $I_{recover}$ . Because there are some image blocks that are not recovered, perform the following operations.

- Step 1* : Calculate the average  $B'_{surround}$  of the surrounding recovered image blocks around the image block  $B'$ .
  - Step 2* : Recover the image block  $B'$  according to  $B'_{surround}$ .
- The above steps (1) to (2) are performed for each unrecovered image blocks to obtain the final recovery image  $I'_{recover}$ .

### 4 Experimental results and analysis

In this paper, we use the  $512 \times 512$  gray images for our experiments. We use the normal images (Peppers, Lena, Plane, Baboon), texture images (Leather, Plastic, Straw) and remote sensing images (Downtown, Island, Shelter) as the test images. Besides, the peak signal to noise ratio and the structure similarity of the image are used to measure the ability of localization and recovery of our algorithm [29].

#### 4.1 Peak signal-to-noise ratio and image structure similarity

##### 4.1.1 Peak signal-to-noise ratio

Assume the images are the reference image  $f$  and the test image  $g$ , whose size is  $M \times N$ , the calculation formula between  $f$  and  $g$  is as follows.

$$MSE(f, g) = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (f_{ij} - g_{ij})^2 \tag{8}$$

$$PSNR(f, g) = 10\log_{10}(255^2/MSE(f, g)) \tag{9}$$

When MSE approaches zero, the PSNR is near infinity, that indicates higher PSNR provides higher image quality. The peak signal-to-noise ratio can reflect the mean square error between the watermark image and the original image.

**Table 3** The effect of recovery on copy attack

Image	Lena	Peppers	Plane	Baboon
PSNR	58.4189	50.7545	53.2910	49.6337
SSIM	0.9996	0.9979	0.9989	0.9970

The larger value shows the smaller difference between the embedded image and the original image.

##### 4.1.2 Image structure similarity

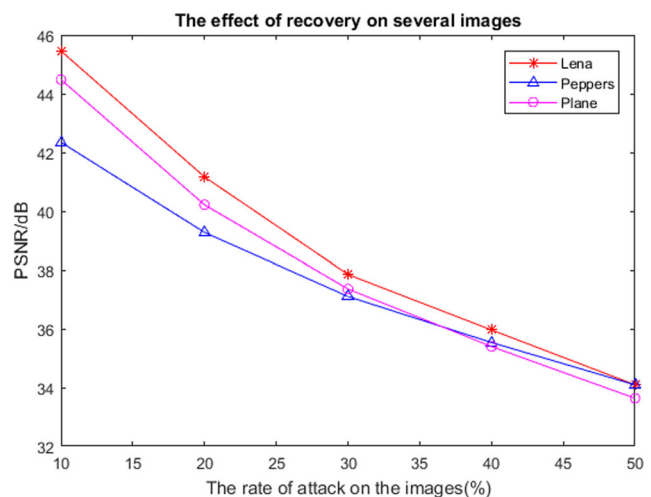
Structured similarity is not designed using a traditional error summation method, but by modeling any image distortion as a combination of three factors, which are correlation loss, luminance distortion, and contrast distortion. Assume the images are the reference image  $f$  and the test image  $g$  whose size is  $M \times N$ , the formula is as follows.

$$\begin{cases} l(f, g) = \frac{2\mu_f\mu_g+C_1}{\mu_f^2+\mu_g^2+C_1} \\ c(f, g) = \frac{2\sigma_f\sigma_g+C_2}{\sigma_f^2+\sigma_g^2+C_2} \\ s(f, g) = \frac{\sigma_{fg}+C_3}{\sigma_f\sigma_g+C_3} \end{cases} \tag{10}$$

$$SSIM(f, g) = l(f, g) \times c(f, g) \times s(f, g) \tag{11}$$

$\mu_f, \mu_g$  are the average of the reference image  $f$  and the test image  $g$ ,  $\sigma_f, \sigma_g$  stand for their standard deviation,  $\sigma_f^2, \sigma_g^2$  are their variance,  $\sigma_{fg}$  is the covariance. In order to avoid the above formula denominator to 0,  $C_1, C_2$  and  $C_3$  are constants. In general,  $C_1 = (K_1 \times L)^2, C_2 = (K_2 \times L)^2, C_3 = C_2/2$ . Usually,  $K_1 = 0.01, K_2 = 0.03, L = 255$ .

We use the peak signal to noise ratio and structured similarity of the image to measure the image quality. Figure 4 shows the original image of Lena, Peppers, Plane and Baboon, and the image after adding watermark. As shown in Table 1, this method increases the PSNR



**Fig. 8** The effect of recovery for different attack





**Fig. 9** The effect of location and recovery

after embedding the watermark, and this paper has great superiority in embedding the watermarking invisibility.

In order to fully validate the effectiveness of our algorithm of our algorithm, we also test the texture images and remote sensing images, whose superiority is shown in Fig. 5. And as shown, the *PSNR* of those images are greater than 47/*dB*, while the *SSIM* of them are greater than 0.989, which is almost closer to 1.

As we can see, the performance of our algorithm has the high peak signal-to-noise ratio and image structure similarity, which reflect the mean square error between the watermark image and the original image. The larger

value shows the smaller difference between the embedded image and the original image. The higher *PSNR* and *SSIM* mean the images embedded into watermark have greater invisibility, which is one of superiorities of our algorithm.

**Table 4** The *PSNR* and *SSIM* of images

Image	Lena	Baboon	Peppers	Plane
<i>PSNR</i>	60.77	57.88	59.55	62.30
<i>SSIM</i>	0.9993	0.9994	0.9993	0.9997

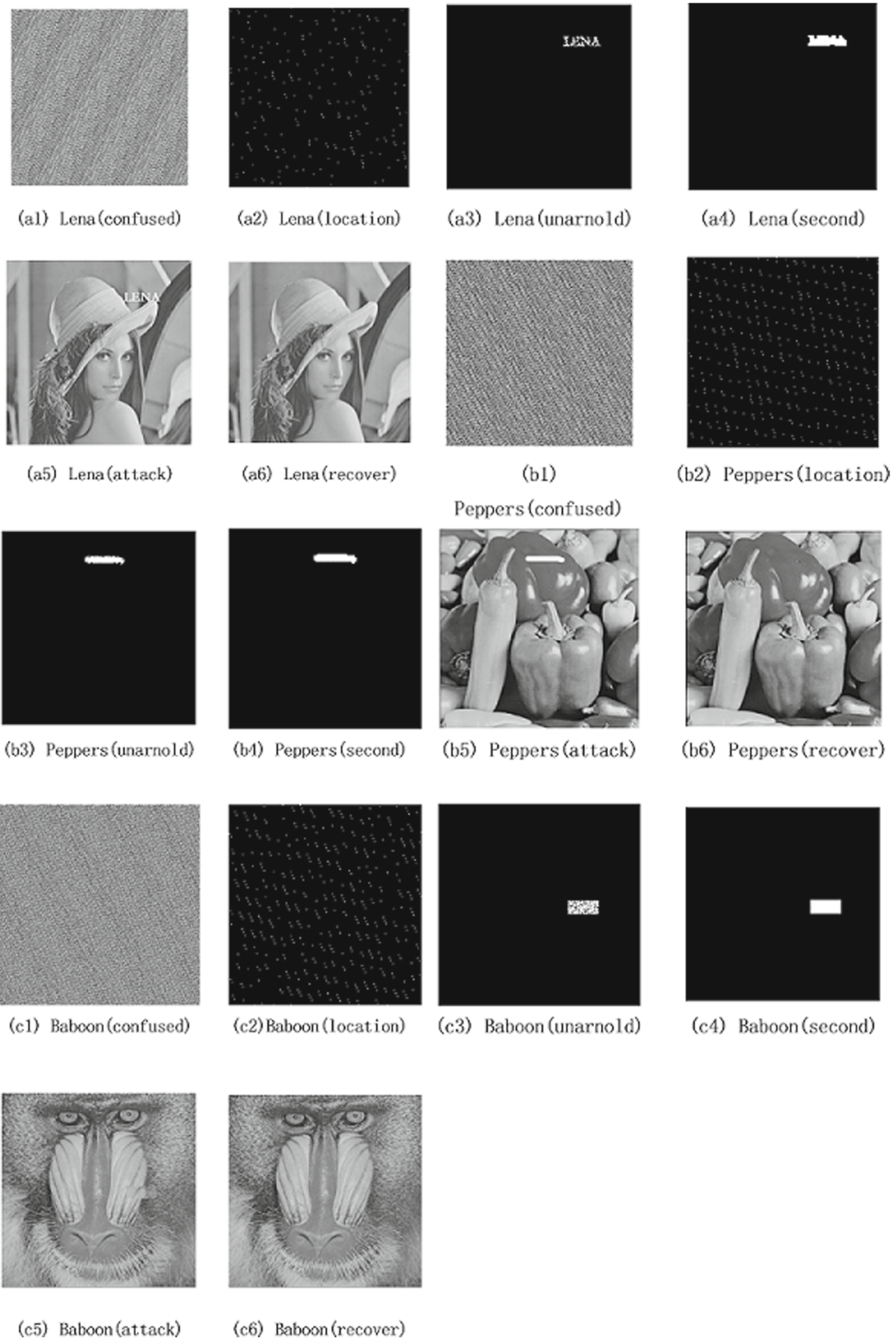


Fig. 10 Location and recovery of scrambled image

## 4.2 Test and analysis

### 4.2.1 Cut attack

As shown in Fig. 6, the normal images are used to test the effectiveness of our method for cut attack. While, (a1), (b1), (c1), (d1) are the images with secure watermark, and (a2), (b2), (c2), (d2) are the tampered image after cut attack. The difference between the attacked images and the normal images is close to half. However, our algorithm can detect the tampers accurately shown in (a3), (b3), (c3), (d3). And (a4), (b4), (c4), (d4) are the recovery results, which shows that this method has a great advantage in resisting attack and the recovery images have high *PSNR* and *SSIM*.

After the shear attack, we can see that the image Lena has the half lost and our algorithm can localize the attack precisely and perfectly. Besides, through recovery of our method, the image has very little difference compared with the original image shown in Table 2, which shows that our algorithm has unparalleled superiority.

### 4.2.2 Copy attack

In this paper, our algorithm propose a authentication watermark with a secure key shown in Section 3, which means every image will have different authentication watermark even though the images have the same pixels. In Fig. 7, (a2), (b2), (c2), (d2) are the copy attacked images, while (a3), (b3), (c3), (d3) are the localization results. Compared with the original images (a1), (b1), (c1), (d1), the recovery images (a4), (b4), (c4), (d4) have a little difference, which suggests our algorithm has great effect of recovery and localization.

When the images are tampered by copy attack, the tampered images can be barely distinguished by the human visual system, but our algorithm can detect the tampers accurately and superbly. Not only that, the excellent effect of the recovery is shown in Table 3, which can be obtained that the *PSNR* is almost more than 50/dB, suggesting the recovery images extremely similar to the original images.

For further experiments, we use the three normal images (Lena, Peppers, Plane) to test the recovery rate of our algorithm with *PSNR*. Figure 8 shows the recovery results when the images experience different attack of varying degrees. As we expect, when the rate of attack increases, the quality of recovery image is becoming worse. However, the *PSNR* of the recovery images is more than 32/dB, even though the image lose the half, which means our algorithm has a large superiority in the process of recovery.

**Table 5** The effect of recovery for scrambled image

Image	Lena	Peppers	Baboon
PSNR	46.5037	45.7986	46.4860
SSIM	0.9787	0.9900	0.9811

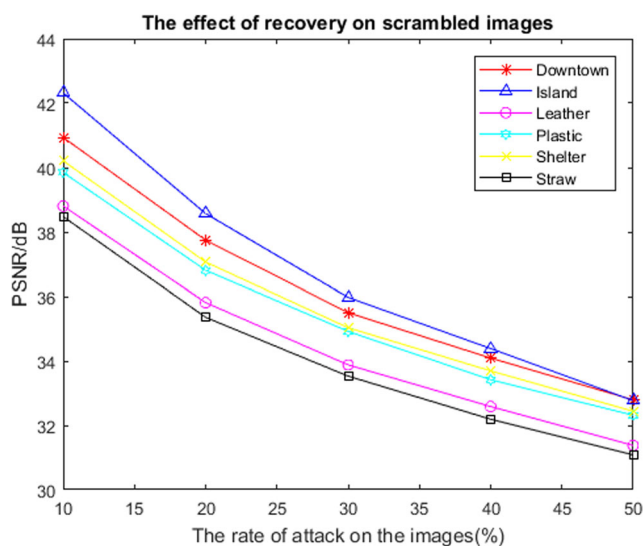
### 4.2.3 Text attack

After the experiments of cut attack and copy attack, we test our algorithm for the detection and recovery. In Fig. 9, (a2), (b2), (c2), (d2) are the tampered images after text attack while (a3), (b3), (c3), (d3) and (a4), (b4), (c4), (d4) are respectively the location results and the recovery images. When the images are suffered the text attack, our algorithm can precisely detect the tampers, and the quality of the recovery images are extremely high in Table 4. The method we propose has a great advantage in resisting attack and recovering from the attacked images.

As the above figure and table shows, the consequences of proposed algorithm has incomparable preponderance in resisting the text attack and recovering from tampered images, the *PSNR* of the recovery images are close to 60/dB, suggesting the recovery images and the original images are almost exactly the same. This advantages in our paper cannot be ignored.

### 4.2.4 Detection and recovery of scrambling image

The algorithm can directly locate the scrambling image, and it has the same effect compared with the plain-image. The experimental results are shown in Fig. 10 and Table 5. As



**Fig. 11** The recovery effect of different scrambling images for various degree attack

we can see, tamper can be localized in the scrambling image and the image can be recovery accurately, that is a great innovation, and the result in the scrambling image is still as well as the plain-image.

To test the effectiveness of our algorithm, we use the texture images (Leather, Plastic, Straw) and remote sensing images (Downtown, Island, Shelter) for further experiments. Shown in Fig. 11, the results of our algorithm in scrambled images are as good as the results in plain-images, which means the tampers can be detected in scrambled images and the recovery results are still good enough compared with plain-images.

## 5 Conclusion

This paper proposes a semi-fragile digital image watermarking algorithm for plain-image and scrambling image. The main features and superiorities of this algorithm include the following aspects:

- (1) The hierarchical idea is used to locate tamper, and it has better anti-shear attack ability, and adds a recovery watermark for tamper recovery. The authentication watermark is composed of the parity check code and the comparison result; the recovery watermark is the average pixel value.
- (2) The secure key is used for authentication watermark, and the attackers can't counterfeit the authentication watermark without the secure key, which vastly increases the security of the watermark and the security of the images [38–40].
- (3) The embedded algorithm uses the well-known spatial domain LSB algorithm. The aim is to improve the tamper recovery effect. The algorithm is simple in principle, but has higher location accuracy and better recovery effect.
- (4) This paper use three layers to detect and locate tamper. In this algorithm, the experimental verification can detect the location of tampering in the image, and can effectively recover the tampering content, and can effectively prevent the vector quantization attack.
- (5) This algorithm can directly locate tamper in the scrambling conditions, and it can detect tamper without revealing the plain-image, greatly improve privacy and security of the image.


**Acknowledgements** This paper is supported by the National Science Foundation of China under grant No.61401060, 61501080, 61572095 and 61771090, the Fundamental Research Funds for the Central Universities' under No. DUT16QY09, and the Social Science Foundation of Jiangxi Province, China No.15JY48.

## References

1. Cox IJ, Kilian J, Leighton FT, Shamoon T (1997) Secure spread spectrum watermarking for multimedia. *IEEE Trans Image Process* 6(12):1673–1687
2. Cox I, Miller M, Bloom J, Fridrich J, Kalker T (2007) *Digital watermarking and steganography*. Morgan Kaufmann, Burlington
3. Song C, Sudirman S, Merabti M (2012) A robust region-adaptive dual image watermarking technique. *J Vis Commun Image Represent* 23(3):549–568
4. Barni M, Bartolini F (2004) *Watermarking systems engineering: enabling digital assets security and other applications*. Marcel Dekker, New York
5. Min W, Bede L (2003) Data hiding in image and video I Fundamental issues and solutions. *IEEE Trans Image Process* 12(6):685–695
6. Watson AB (1993) DCT Quantization matrices visually optimized for individual images. *Proc SPIE Human Vis Vis Process Digit Display* 9:202–216
7. Mohan BC, Kumar SS (2008) Robust digital watermarking scheme using contourlet transform. *Int J Comput Sci Netw Security* 8(2):43–51
8. Giakoumaki A, Pavlopoulos S, Koutouris D (2003) A medical image watermarking scheme based on wavelet transform. In: *Proc 25th Annu Int Conf IEEE Eng Med Biol Soc*, pp 856–859
9. Sadreazami H, Ahmad MO, Swamy MNS (2014) A study of multiplicative watermark detection in the contourlet domain using alpha-stable distributions. *IEEE Trans Image Process* 23(10):4348–4360
10. Karybali IG, Berberidis K (2006) Efficient spatial image watermarking via new perceptual masking and blind detection schemes. *IEEE Trans Inf Forensics Security* 1(2):256–274
11. Zeki AM, Manaf AA (2011) ISB Watermarking embedding: block based model. *Inf Technol J* 10(4):841–848
12. Emami MS, Sulong GB, Seliman SB (2012) An approximation approach for digital image owner identification using histogram intersection technique. *Int J Innov Comput Inf Control* 8(7):4605–4620
13. Wang Y, Pearmain A (2004) Blind image data hiding based on self reference. *Pattern Recogn Lett* 25(15):1681–1689
14. Bhatnagar G, Wu QMJ, Raman B (2012) A new robust adjustable logo watermarking scheme. *Comput Secur* 31(1):40–58
15. Patra JC, Phua JE, Bornand C (2010) A novel DCT domain CRT-based watermarking scheme for image authentication surviving JPEG compression. *Digit Signal Process* 20(6):1597–1611
16. Su Q, Niu Y, Zou H, Liu X (2013) A blind dual color images watermarking based on singular value decomposition. *Appl Math Comput* 219(16):8455–8466
17. Su Q, Niu Y, Wang G, Jia S, Yue J (2014) Color image blind watermarking scheme based on QR decomposition. *Signal Process* 94:219–235
18. Zhu X, Ding J, Dong H, Hu K, Zhang X (2014) Normalized correlation-based quantization modulation for robust watermarking. *IEEE Trans Multimedia* 16(7):1888–1904
19. Horng S-J, Rosiyadi D, Li T, Takao T, Guo M, Khan MK (2013) A blind image copyright protection scheme for e-government. *J Vis Commun Image Represent* 24(7):1099–1105
20. Shen J-J, Hsu P-W (2007) A robust associative watermarking technique based on similarity diagrams. *Pattern Recognit* 40(4):1355–1367
21. Andalibi M, Chandler DM (2015) Digital image watermarking via adaptive logo texturization. *IEEE Trans Image Process* 24(12):5060–5073

22. Barni M, Bartolini F, Piva A (2001) Improved wavelet-based watermarking through pixel-wise masking. *IEEE Trans Image Process* 10(5):783–791
23. Chang CS, Shen JJ (2017) Features classification forest: a novel development that is adaptable to robust blind watermarking techniques[J]. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society PP(99)*:1–1
24. Urvoy M, Goudia D, Atrousseau F (2014) Perceptual DFT watermarking with improved detection and robustness to geometrical distortions[J]. *IEEE Trans Inf Forensics Secur* 9(7):1108–1119
25. Voloshynovskiy S, Pereira S, Pun T, Eggers JJ, Su JK (2001) Attacks on digital watermarks: classification, estimation based attacks, and benchmarks. *IEEE Commun Mag* 39(8):118–126
26. Blum L, Blum M, Shub M (1986) A Simple unpredictable pseudorandom number generator. *SIAM J Comput* 15(2):364–383
27. Celik MU (2002) Hierarchical watermarking for secure image authentication with localization [J]. *IEEE Trans Image Process.* 11(6):585–595
28. Wong PW, Memon N (2002) Secret and public key authentication schemes that resist vector quantization attack[A]. In: *Proc of SPIE security and watermarking of multimedia[C]*. San Jose, USA, pp 1593–1601
29. Hore A, Ziou D (2010) Image quality PSNR vs. SSIM[c]. In: *International conference on pattern recognition, IEEE*, 2366–2369
30. Huang DZ, Chen ZG, Zhu CX (2008) Image spatial domain watermark algorithm based on general Arnold mapping and neural network[J]. *Appl Res Comput* 25(4):1144–1146
31. Shao L, Qin Z, Xingchen H (2007) Algorithm of spatial domain image watermark based on scrambling transformation of image[J]. *Comput Eng* 33(2):122–124
32. Celik MU et al (2002) Hierarchical watermarking for secure image authentication with localization. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society* 11(6):585
33. Cox IJ, Kilian J, Leighton FT et al (1997) Secure spread spectrum watermarking for multimedia[J]. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society* 6(12):1673–1687
34. Rota GC (1990) *Geometrical methods in the theory of ordinary differential equations* : V. I. Arnold, 2nd ed. Springer, 1988, 351 pp[J]. *Adv Mater* 80(2):269–269
35. Qiu T, Zhao A, Xia F, Si W, Wu. DO (2017) ROSE: Robustness strategy for scale-free wireless sensor networks. *IEEE/ACM Trans Netw* 25(5):2944–2959
36. Qiu T, Qiao R, Han M, Sangaiah AK, Lee I (2017) A lifetime-enhanced data collecting scheme for internet of things. *IEEE Commun Mag* 55(11):132–137
37. Qiu T, Zheng K, Han M, Philip Chen CL, Xu M (2017) A data-emergency-aware scheduling scheme for internet of things in smart cities. *IEEE Transactions on Industrial Informatics.* <https://doi.org/10.1109/TII.2017.2763971>
38. Guo C, Chen X, Jie YM et al (2017) Dynamic multi-phrase ranked search over encrypted data with symmetric searchable encryption. *IEEE Transactions on Services Computing*, published online. <https://doi.org/10.1109/TSC.2017.2768045>
39. Guo C, Luo NQ, Alam Bhuiyan MZ et al (2017) Key-aggregate authentication cryptosystem for data sharing in dynamic cloud storage. *Future Generation Computer Systems*, published online. <https://doi.org/10.1016/j.future.2017.07.038>
40. Guo C, Zhuang RH, Jie YM et al (2016) Fine-grained database field search using attribute-based encryption for E-Healthcare clouds. *J Med Syst* 40(11):235;1–235:8

## Affiliations

Bin Feng<sup>1</sup> · Xiangli Li<sup>2,3</sup> · Yingmo Jie<sup>2,3</sup> · Cheng Guo<sup>2,3</sup>  · Huijuan Fu<sup>4,5</sup>

Yingmo Jie  
jymsf2015@mail.dlut.edu.cn

Huijuan Fu  
huijuanfu@163.com

<sup>1</sup> School of Information Science and Technology, Taishan University, Taian, 271000, China

<sup>2</sup> School of Software Technology, Dalian University of Technology, Dalian, 116620, People's Republic of China

<sup>3</sup> Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian, 116620, China

<sup>4</sup> School of Information Management, Wuhan University, Wuhan, 430014, People's Republic of China

<sup>5</sup> School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou, 341000, People's Republic of China