



ELSEVIER

Pattern Recognition Letters 23 (2002) 931–941

Pattern Recognition
Letters

www.elsevier.com/locate/patrec

An image intellectual property protection scheme for gray-level images using visual secret sharing strategy

Chin-Chen Chang^{a,*}, Jun-Chou Chuang^a

^a *Department of Computer Science and Information Engineering, National Chung Cheng University, Minghsiang, Chiayi 621, Taiwan, ROC*

Received 18 May 2001; received in revised form 13 September 2001

Abstract

In this paper, we describe an image protection method for image intellectual property. The concept of this paper is based on the visual secret sharing scheme. We use the visual secret sharing scheme to construct two shares (also called transparencies). One share is generated from the host image, and the other share is arbitrarily generated by the owner. When the owner needs to be identified, we can superimpose these two shares to recover the secret information in order to validate the ownership. Our method has three advantages. The first advantage is simple computation, which means that our method does not use any complex computations such as discrete cosine transformation (DCT), and thus our method can be executed efficiently. The second advantage is that our method does not change any pixels of the host image. The third advantage is that our method is secure and robust. From the experimental results, our method can tolerate several image attacks, e.g., blurring, cropping, sharpening, JPEG lossy compression, etc. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Torus automorphism; Visual cryptography; Watermarking; Discrete cosine transformation; Verification

1. Introduction

The Internet has been rapidly gaining popularity and from it, people can quickly and easily obtain many kinds of data from the Internet such as images, texts, music, pictures, etc. However, some kinds of data cannot be arbitrarily copied or distributed. Therefore, copyright protection has become a very important research topic (Kundur and

Hatzinakos, 1999). The most frequently used data are images and pictures.

Watermarking is a kind of image protection method that embeds watermarks into the host image. The watermark is often embedded into the frequency domain rather than into the spatial domain. This occurs because watermarks embedded into the frequency domain can be more robust. Two transformation techniques, discrete cosine transformation (DCT) (Hsu and Wu, 1999) and wavelet transformation (Kim et al., 1999), are used frequently. However, both techniques are time consuming. They also change some pixels of the host image, thus decreasing its image quality.

* Corresponding author. Tel.: +886-5-272-0411x6011; fax: +886-5-272-0859.

E-mail addresses: ccc@cs.ccu.edu.tw (C.-C. Chang), lzchung@cs.ccu.edu.tw (J.-C. Chuang).

Recently, Hou and Chen (2000) proposed an asymmetric watermarking scheme that is based on the visual secret sharing scheme. They used Naor and Shamir's (1994) visual secret sharing scheme to split the watermark into two shares referred to as share-1 and share-2. Like a binary image, share-1 and share-2 can be treated as random noise. Share-1 is embedded into the host image using pixel value decrease or increase. Share-2 is used as the secret key. The original image and the secret key can be used to extract the embedded watermark from the host image to verify the ownership of an image. The main drawback of their method is that it modifies the host image and also requires the original image. Our method does not. In general, most watermarking techniques do not require the original image.

With its improved execution efficiency and good image quality, a new image protection method based on the visual secret sharing scheme is proposed. Our method combines two techniques. They are the torus automorphism technique (Voyatzis and Pitas, 1996) and the visual secret sharing scheme (Chang and Hwang, 1998; Hwang and Chang, 1999, 2001). The torus automorphism technique is used to scramble each pixel of the host image. Then, we apply Hwang and Chang's (2001) visual secret sharing scheme to generate two shares. The visual secret sharing technique uses the human visual system of pixel density to retrieve the information. The main difference between Naor and Shamir's and Hwang and Chang's (2001) methods is found in whether or not the share is meaningful. In Naor and Shamir's scheme, each share looks like random noise and it does not contain anything meaningful. In (Hwang and Chang's (2001) scheme), each share can be designed to be more meaningful by adding such information as the author name, logos, words, etc.

Our method applies the torus automorphism to scramble each pixel of the host image. The result is a chaotic mixing image. We randomly select a sub-image from the chaotic mixing image. The sub-image is divided into many blocks and uses a sorting algorithm to sort each block. The gray-level sub-image uses the sorting algorithm to convert it into a binary image. Thus share-1 is generated. Next, we apply Hwang and Chang's

visual secret sharing scheme (Hwang and Chang, 2001) to generate share-2. The secret image is hidden into share-1 and share-2. We do not want to change the original image, thus no pixel from share-1 will be changed. So, we need to adjust some pixels in share-2 so that the secret image can be hidden in two shares. Further, we can superimpose these two shares to extract the ownership of the image. Our method is based on Hwang and Chang's visual secret sharing scheme (Hwang and Chang, 2001) that is perfectly secure and simple. Our method does not require complex computation and does not modify any pixel from the original image. Our method is practically suitable for the application in which the modification of the lost image is not allowed. For example, many medical images are not allowed to have any modifications of them. In this case, our image intellectual property protection scheme fits this requirement.

The rest of this paper is organized as follows. Section 2 introduces work relating to two visual secret sharing schemes. The details of our method are proposed in Section 3. Section 4 presents some experimental results and analysis. Finally, the conclusions are presented in Section 5.

2. Previous work

2.1. Review of Naor and Shamir's visual secret sharing scheme

Naor and Shamir (1994) proposed a visual secret sharing scheme (also referred to as visual cryptography). The main advantage of their method is simple computation and security. The conventional cryptographic system requires more complex computations.

We briefly describe Naor and Shamir's visual secret sharing scheme and list the stacking results in Fig. 1. The visual secret sharing technique is only suitable for the binary image. There exists a secret image with a size of $M \times N$. The secret image is composed of share-1 and share-2. The image size of the two shares is $2M \times 2N$. Share-1 and share-2 are divided into two sets of non-overlapping blocks B^1 and B^2 , respectively, where each block in B^i contains 2×2 pixels. We choose a

Image	Secret pixel (white)	Secret pixel (black)
Share-1		
Share-2		
Stacking result		

Fig. 1. All cases of the stacking results of Noar and Shamir's scheme.

block B_i^1 from B^1 and another block B_i^2 from B^2 in the same position to represent a secret pixel. That is to say, each secret pixel uses two 2×2 pixels to represent it. Here, each block contains half white pixels and half black pixels. If a secret pixel is black in color, then each pixel in B_i^1 is complementary to each pixel in B_i^2 . Otherwise, the positions of the black and white pixels in B_i^1 and B_i^2 are the same.

Noar and Shamir's scheme is secure because without possessing both shares, it is very difficult for anyone who wants to retrieve the secret image from each share to do so. The main drawback of their scheme is that each share looks like random noise and does not have any meaning. Therefore, Hwang and Chang (2001) proposed a new scheme to improve upon Noar and Shamir's scheme.

2.2. Review of Hwang and Chang's visual secret sharing scheme (Hwang and Chang, 2001)

Procedurally, Hwang and Chang's visual secret sharing scheme (Hwang and Chang, 2001) is similar to Noar and Shamir's visual secret sharing scheme. The main difference is in the meaningfulness of each share. In Noar and Shamir's scheme, each share looks like random noise and does not have any meaning. But in Hwang and Chang's scheme, each share can be designed to contain more meaning. Therefore, designing a meaningful share is the key challenge.

In Hwang and Chang's scheme, each block in share-1 and share-2 contains 3×3 pixels. If we want to generate a meaningful share, then we must define some rules as to the meaning of the black-

ness and whiteness of a block. Whether the block is black or white is dependent upon using the human visual system to detect color according to the color pixel density of a block. That is to say that a block that contains more black pixels than white pixels has black as its meaning. Otherwise, the block has white as its meaning. So, we can arbitrarily design meaningful images on each share. We give an example of the black and white definition of share-1, share-2, and the stacking results in Fig. 2. Here (t,s) has the meaning of a block that contains t black pixels and s white pixels. In Fig. 2, a block in share-1 and share-2 has a meaning of black if this block contains 7 black pixels and 2 white pixels. Otherwise, this block contains 5 black pixels and 4 white pixels. If the stacking result from share-1 superimposed on share-2 of a block contains 8 black pixels and 1 white pixel, it has a meaning of white. Otherwise, the stack result of this block contains 9 black pixels and 0 white pixels. The other procedures of Hwang and Chang's scheme are the same as in Noar and Shamir's scheme.

The security of Hwang and Chang's scheme is the same as that of Noar and Shamir's scheme and each share can be made more meaningful.

Image	White	Black
Share-1 and share-2	(5,4)	(7,2)
Stacking result	(7,2)	(9,0)

Fig. 2. An example of the black and white definition of a block.

Our method applies Hwang and Chang's scheme (2001) to generate the two shares.

3. Proposed method

3.1. Signing procedure

Our signing procedure contains two parts. The first part is the scramble, and the second part is the share construction. First, we apply the torus automorphism (Voyatzis and Pitas, 1996) to scramble all pixels of the host image. Next, we apply Chang and Hwang's visual secret sharing scheme (Hwang and Chang, 2001) to construct two shares. The flow chart of our proposed method is shown in Fig. 3.

There exists a gray-level image O , with a size of $N \times M$ pixels. We apply the torus automorphism to scramble each pixel of the image O . The formula is listed as

$$\begin{pmatrix} X_{i+1} \\ Y_{i+1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ k & k+1 \end{pmatrix} \begin{pmatrix} X_i \\ Y_i \end{pmatrix} (\text{mod } N), \quad (1)$$

where (X_i, Y_i) is a coordinate in two-dimensional space, and the variable i is the number of sequences of the torus automorphism. As for N and k , they are denoted as the size of the given image and the secret key, respectively. The secret key k can be arbitrarily chosen by the owner. For example, assume a pixel p is stored in coordinate

$(2, 9)$. After i sequences of the torus automorphism, the pixel p will be stored in the new coordinate (x', y') . The relational positions of pixels in the host image will be changed by the torus automorphism. After finishing the above work, we obtain a chaotic mixing image O' . Here the size of the chaotic mixing image is the same as the host image O .

In the next part, we generate two shares using Hwang and Chang's scheme. There exists a binary secret image logo-2 with a size of $N' \times M'$. The secret image logo-2 is hidden in two shares. They are share-1 and share-2. Before generating the share-1 and share-2, we need to define the blackness and whiteness of a block in share-1, share-2 as well as the stacking results. Some details of the black and white definition were described in the previous section. Assume the black and white definition in share-1, share-2 and the stacking results of a block are $\{(9-t, t), (9-t, t)\}$, $\{(d, 9-d), (h, 9-h)\}$, and $\{(e, 9-e), (f, 9-f)\}$, respectively.

We can obtain share-1 in the following way. First, we randomly choose a coordinate (x, y) in O' , and then we select a sub-image I from O' . The size of the sub-image is $3N' \times 3M'$ pixels. Here $N \geq 3N'$ and $M \geq 3M'$. Next, we need to add a step to convert the sub-image I into a binary image. This occurs because the visual secret sharing scheme is only suitable for binary images. We divide the sub-image into many non-overlapping blocks, where each block contains 3×3 pixels.

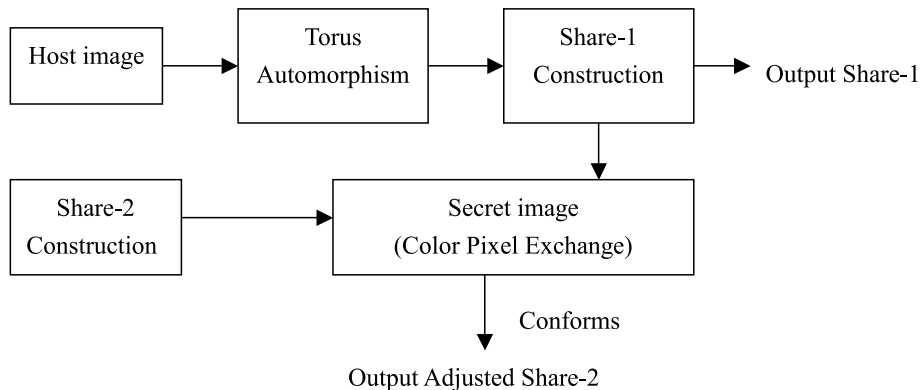


Fig. 3. The flow chart of our proposed method.

We use a sorting algorithm to sort 3×3 pixels in each block, and the smallest t pixels in each block are chosen. The smallest t pixels in each block are set to be white, and the other $(9 - t)$ pixels are set to be black. Therefore, share-1 is generated.

Share-1 is generated from the host image and it does not have any meaning. Share-2 in our method can be made more meaningful and can be used as the secret key. We then input a meaningful binary image logo-1, e.g. logos, numbers, authors' names, etc. Its size is $N' \times M'$ pixels. The image size of the meaningful image is the same as the secret image. We extend each pixel in the meaningful image to 3×3 pixels a block. If a pixel from the meaningful image is black pixel, then its block contains d black pixels and $(9 - d)$ white pixels. Otherwise, its block contains h black pixels and $(9 - h)$ white pixels. A block that contains more black pixels than white pixels has a meaning of black. Otherwise, the block has a meaning of white. Therefore, the value d is larger than h . Finally, we obtain a binary share called share-2 and its size is the same as share-1.

The secret image logo-2 is hidden in share-1 and share-2. Therefore, in the next step, we adjust share-1 and share-2 to hide the secret image logo-2. We do not want to change the image quality of the host image. Therefore, share-1 does not need to be changed. We change some pixels in share-2, but we do not destroy the number of black pixels and white pixels of a block. If a secret pixel in S is black or white, then it uses the black and white definition of a block $(e, 9 - e)$ and $(f, 9 - f)$ to fill the corresponding black and white pixels. Here the value e is larger than f . If the stacking result does not satisfy the above definition, then we shall adjust some pixels in share-2 using the color pixel exchange method. We can exchange some black pixels and white pixels in a block, but the number of black pixels and white pixels must be equal to the predefined numbers. Therefore, the secret image is hidden in share-1 and share-2.

Finally, two shares are generated. We store secret image logo-2, share-2, three parameters i , k , and N , and the coordinate (x, y) and the black and white definition table in the certification authority (CA). Logo-1 and the original image can be published.

Signing Procedure Algorithm

Input. A gray-level host image O with $N \times M$ pixels.

Output. Two shares, share-1 and share-2.

- *Step 1.* Use the torus automorphism to scramble the host image and then obtain a chaotic mixing image O' .
- *Step 2.* Share-1 generation steps.

Step 2.1. Choose a coordinate (x, y) and obtain a sub-image I from O' . The image size of I is $3N' \times 3M'$.

Step 2.2. Divide I into many small blocks with each block containing 3×3 pixels.

Step 2.3. Use a sorting algorithm to sort each block. The smallest t pixels are set as white, and the other $(9 - t)$ pixels are set as black.

- *Step 3.* Apply Hwang and Chang's visual secret sharing scheme to construct a meaningful share called share-2 that is the same size as share-1.

Step 3.1. Read a meaningful image logo-1. Its size is $N' \times M'$.

Step 3.2. Each pixel of logo-1 is extended to 3×3 pixels a block, and each block is treated according to the black and white definition in share-2 to generate share-2.

- *Step 4.* Hide the secret image logo-2 in share-1 and share-2.

Step 4.1. Read a secret image logo-2. Its size is $N' \times M'$.

Step 4.2. Retrieve a secret pixel (black/white) and the corresponding block with 3×3 pixels in share-1 and share-2.

Step 4.3. Superimpose these two blocks. If the stacking result does not satisfy the black and white definition of stacking results, then we exchange some black pixels with white pixels in share-2 until the result conforms. Otherwise, there is no change to the pixels in share-2.

Step 4.4. Repeat Step 4.2 and Step 4.3 until all secret pixels are processed.

We give an example to demonstrate the color pixel exchange method in Figs. 4 and 5. In Fig. 4, we first give the definition of a black and white

Image	White	Black
Share-1	(8,1)	(8,1)
Share-2	(3,6)	(7,2)
Stacking result	(8,1)	(9,0)

Fig. 4. The black and white definition of a block in our method.

block. In Fig. 5(a), we assume that a secret pixel is black, and then the stacking result from share-1 superimposed on share-2 of a block should be (9,0). However, the stacking result in Fig. 5(a) is not (9,0), thus we need to modify some pixels in share-2 such that the stacking result is satisfied (9,0). Share-1 does not need to be changed. We exchange a black pixel and a white pixel in share-2. The total number of black and white pixels is the same as the predefined number. The exchange result is listed in Fig. 5(b). The other color pixel exchange procedures are listed in Figs. 5(c) and (d).

3.2. Verification procedure

The verification procedure consists of two parts. The first part scrambles each pixel of the host image using the torus automorphism and then retrieves share-1 from the host image. Here, the first part is the same as in the signing procedure. The second part involves the retrieval of meaningful information by superimposing share-1 and share-2. Finally, the retrieved secret information can be identified using the human visual system.

Verification Procedure Algorithm

Input. Host image O , share-2, i, k, N , coordinate (x, y) , the black and white definition table, and the secret image logo-2.

Output. “Legal” or “Illegal”.

- *Step 1.* Use Step 1 and Step 2 of the signing procedure algorithm to generate share-1.

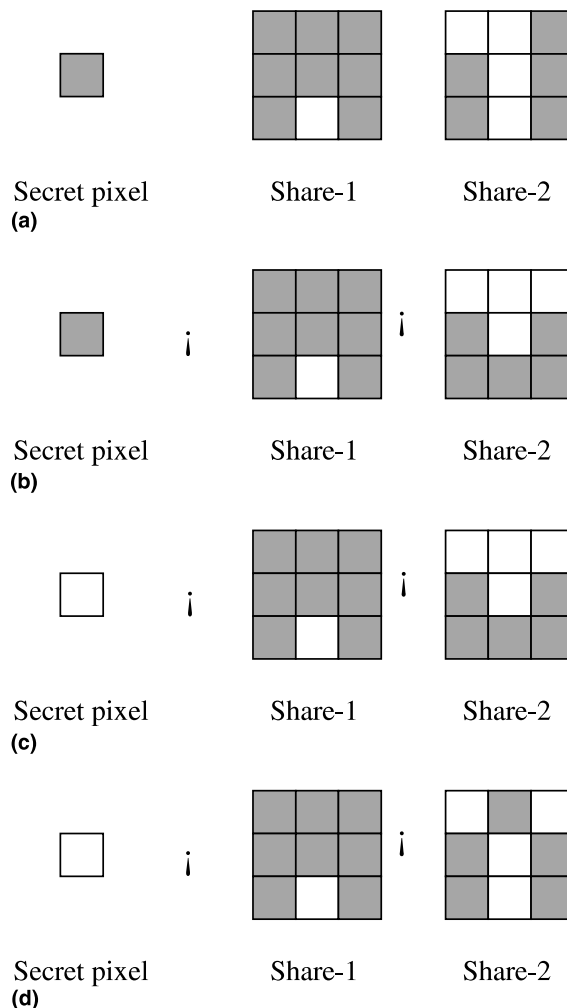


Fig. 5. An example of the color pixel exchange method: (a) case 1; (b) apply color pixel exchange in case 1; (c) case 2; (d) apply color pixel exchange in case 2.

- *Step 2.* Superimpose share-1 and share-2 to retrieve the secret image S' .
- *Step 3.* According to the human visual system, if S' is visually similar to logo-2 then return “Legal”; Otherwise return “Illegal”.

4. Experimental results and analysis

Our experiments were executed on an IBM personal computer (Intel Pentium 300 MHz) using C++ Builder 4.0 programming language. We used

gray-level image “Lena” with a size of 512×512 pixels as our test image. The original image “Lena” and two logos with a size of 64×64 pixels are shown in Fig. 6.

The definition of black and white in share-1 and share-2 and the stacking results of our experiment are listed in Fig. 4. Our method uses the torus automorphism to scramble each pixel of the host image. Three parameters k, N, i of the torus automorphism are set to be 3, 511, 10, respectively. After scrambling, the host image becomes a chaotic mixing image. Next, we randomly select a sub-image from the chaotic mixing image. The image size of the sub-image is $(64 \times 3) \times (64 \times 3)$ pixels. We divide the sub-image into many non-overlapping blocks where each block contains 3×3 pixels. We use a sorting algorithm to sort each block, and then select the smallest element from each

block. The smallest pixel value is set to be white, and the other pixels are set to be black. Share-1 is generated and shown in Fig. 7(a). Here the black and white definition of a block in share-1 is set to be (8, 1) and (8, 1). There are 8 black pixels and 1 white pixel in each block of share-1. We select a meaningful logo in Fig. 6(b). Each pixel in Fig. 6(b) is extended to 3×3 pixels a block, and each block is treated according to the black and white definition in share-2 to generate share-2. Here the black and white definition in share-2 is set to be (7, 2) and (3, 6). The secret image is hidden in share-1 and share-2. The secret image logo-2 is shown in Fig. 6(c). Here the black and white definition in the stacking results is set to be (9, 0) and (8, 1). The share-1 and the adjusted share-2 are listed in Fig. 7(a) and (b). In Fig. 7(c), we list the stacking result of the secret image after superimposing share-1 and share-2.

We perform several image attacks such as blurring, sharpening, JPEG lossy compression (80% compression factor), cropping (10% image cropping), and scaling (scale the image size from 512×512 to 256×256 pixels) on the test image “Lena”. We used the image-processing tool “Ulead PhotoImpact 5.0” to implement the above image attacks. The experimental results are listed in Fig. 8. In Fig. 8(m), we use another image “Tiffany” to retrieve our secret image. However, it could not retrieve the secret image because image “Tiffany” is not our image. Even if each pixel of the gray value of the two images “Lena” and “Tiffany” are similar, our scheme can still recognize these two images. In Table 1, we list the accuracy rate of share-1 under various attacks. The accuracy rate is defined as

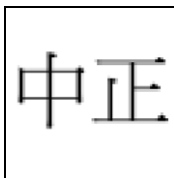
$$\text{Accuracy rate} = \frac{CB}{NB}, \quad (2)$$

where NB is the number of blocks in share-1 and CB is the number of the correct blocks in share-1 after the image attack. If the t smallest elements of a block B_1 in share-1 after image attack are the same as the original block B_1 then add one for CB. Otherwise, we do not decrease or increase CB.

Our scheme has the same security as Hwang and Chang’s visual secret sharing scheme, and it is based on the torus automorphism and the visual



(a)



(b)



(c)

Fig. 6. The original image “Lena” and two logos: (a) test image “Lena” (512×512 pixels); (b) logo-1 (64×64 pixels); (c) logo-2 (64×64 pixels).

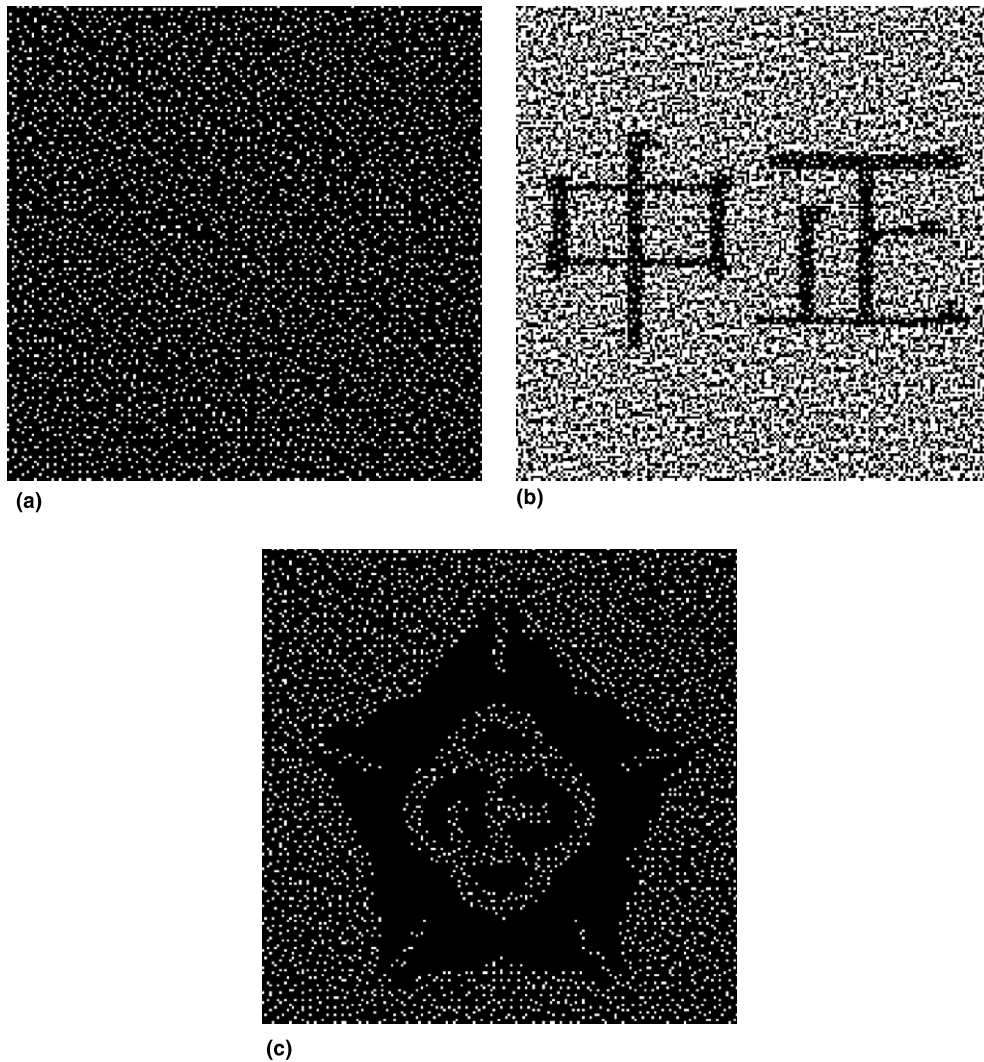


Fig. 7. The results of share-1, share-2, and the retrieved secret image: (a) share-1; (b) share-2; (c) the retrieved secret image from share-1 superimposed on share-2.

secret sharing scheme. Without the three parameters i , N , and k , and share-2, anyone who wants to retrieve the secret image has much difficulty. On the other hand, our scheme randomly selects a sub-image from the chaotic mixing image to generate share-1. Therefore, it is very difficult to guess which pixels of the host image were used. Even though illegal users created share-2, it was very difficult to retrieve the secret image. Without the correct shares, anyone who wants to stack share-1 with share-2 to obtain the secret image has great

difficulty. Share-2 and secret image logo-2 were stored in CA. Logo-1 and original image can be published.

The image robustness of our scheme is dependent upon the torus automorphism and the sorting algorithm. It is difficult to achieve image robustness if we directly generate share-1 from the host image. Due to the fact that each pixel value of the host image is closer to its neighboring pixels, thus we need to scramble each pixel's correlation. There is a good property for the chaotic image. We as-

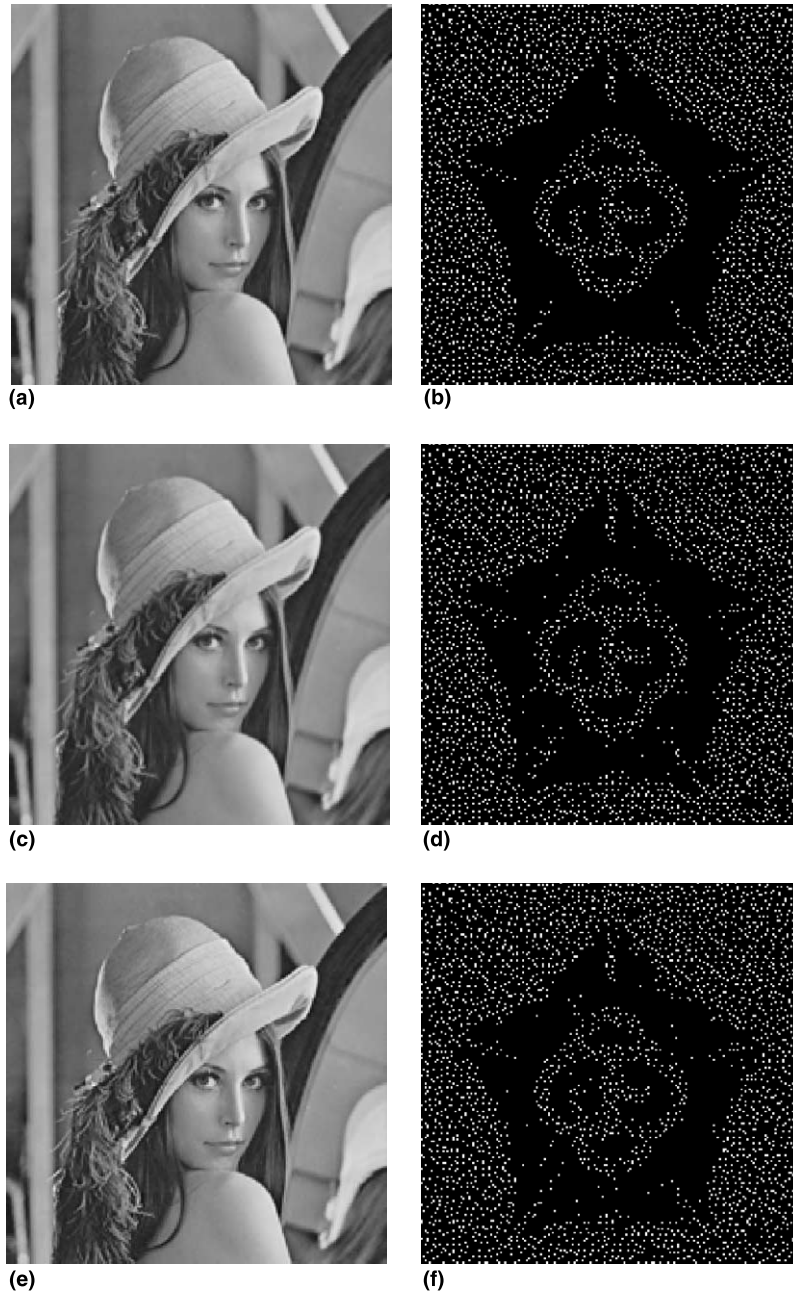


Fig. 8. Several image attacks on images “Lena” and “Tiffany” and the corresponding retrieved secret images: (a) original image “Lena”; (b) the retrieved secret image; (c) blurred image “Lena”; (d) the retrieved secret image; (e) sharpened image “Lena”; (f) the retrieved secret image; (g) JPEG compressed image “Lena” (80% compressed factor); (h) the retrieved secret image; (i) cropped image “Lena” (10% cropping); (j) the retrieved secret image; (k) scaled image “Lena” (512×512 to 256×256); (l) the retrieved secret image; (m) image “Tiffany”; (n) the retrieved secret image.

sume p_1 and p_2 are two pixels in the chaotic image. The pixel value p_1 is larger than pixel value p_2 . If

the host image is not modified too much then pixel p_1 is still larger than pixel p_2 . Therefore, we use a

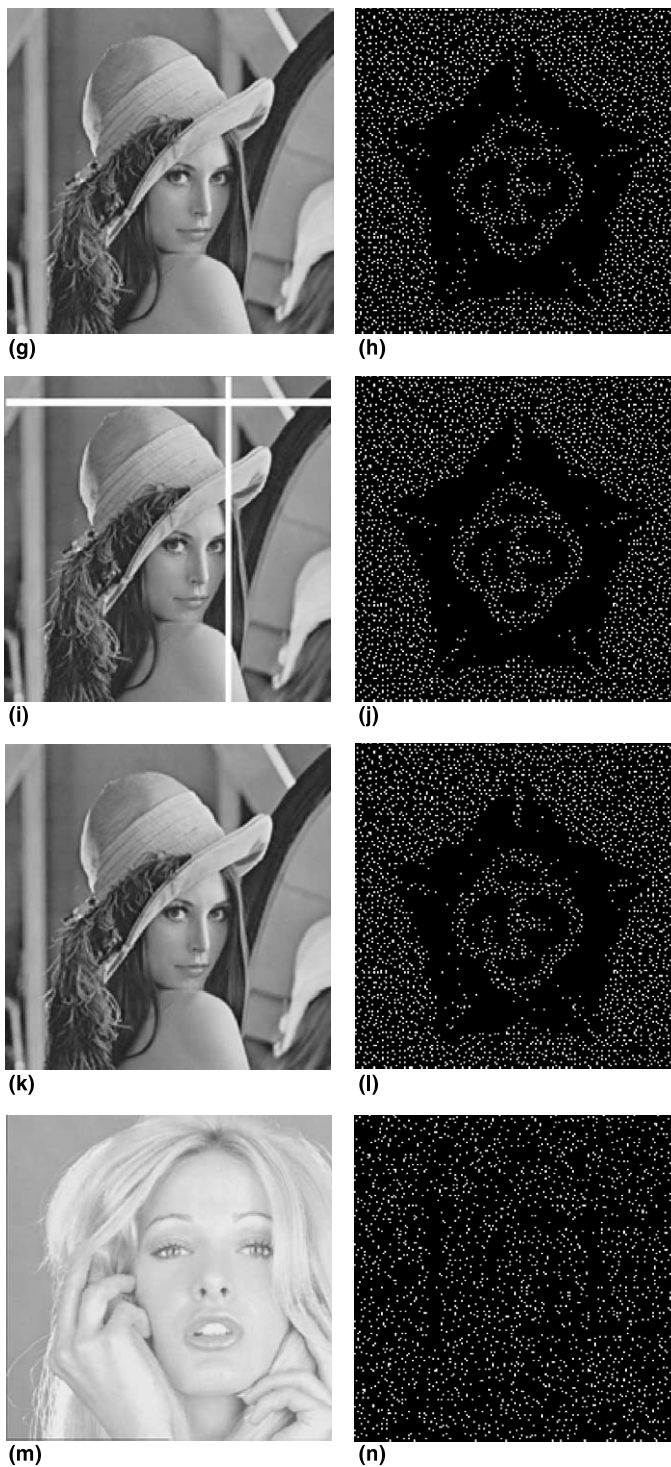


Fig. 8 (continued).

Table 1
The rate of accuracy of share-1 under various attacks

Various attacks	Lena (%)
Blurring	88.72
Sharpening	89.55
JPEG compression (compression factor 80)	91.04
Cropping	95.97
Scaling	88.30

sorting algorithm to sort each block and select the p smallest elements in each block.

According to our experimental results, after image attacks, our scheme is still robust. Our scheme is secure and requires few computations. Our scheme can be applied to unchangeable images, multi-owner host image sharing, mobile phone system, cable TV verification system, smart card, etc.

5. Conclusions

In this paper, we propose a meaningful image protection method that is efficient and effective. It has the following advantages: (1) it does not require complex computations, (2) it does not modify the host image, (3) it is secure and robust, (4) it allows multiple users share one image. Our

method can be applied to unchangeable images, mobile phone verification systems, personal digital assistant (PDA) verification systems, low power verification systems, smart cards, etc.

References

- Chang, C.C., Hwang, R.J., 1998. A simple picture hiding scheme. *Comput. Process. Orient. Lang.* 12, 237–248.
- Hou, C.Y., Chen, M.P., 2000. An asymmetric watermarking scheme based on visual cryptography. In: *Proc. Fifth Signal Process. Conf.*, vol. 2, pp. 992–995.
- Hsu, C.T., Wu, J.L., 1999. Hidden digital watermarks in images. *IEEE Trans. Image Process.* 8, 58–68.
- Hwang, R.J., Chang, C.C., 1999. An improved threshold scheme based on modular arithmetic. *J. Inform. Sci. Eng.* 15, 691–699.
- Hwang, R.J., Chang, C.C., 2001. Hiding a picture in two pictures. *Opt. Eng.* 40, 342–351.
- Kim, W.S., Hyung, O.H., Park, R.H., 1999. Wavelet based watermarking method for digital images using the human visual system. *Electron. Lett.* 35, 466–468.
- Kundur, D., Hatzinakos, D., 1999. Digital watermarking for telltale tamper proofing and authentication. *Proc. IEEE* 87, 1167–1180.
- Naor, N., Shamir, A., 1994. Visual Cryptography, *Advances in Cryptography: Eurocrypt '94*. In: *Lecture Note in Computer Science*. Springer, Perugia, Italy, pp. 1–12.
- Voyatzis, G., Pitas, I., 1996. Applications of total automorphisms in image watermarking. *Image Process. Conf.* 2, 237–240.