

Sharing a Secret Image in Binary Images with Verification

Zhi-hui Wang

School of Software
Dalian University of Technology
Dalian, Liaoning, China.
wangzhihui1017@yahoo.com

Chin-Chen Chang, Huynh Ngoc Tu

Department of Information Engineering and Computer Science
Feng Chia University
Taichung 40724, Taiwan, R.O.C.
ccc@cs.ccu.edu.tw; ngoctu84vn@gmail.com

Ming-Chu Li

School of Software
Dalian University of Technology
Dalian, Liaoning, China.
li_mingchu@yahoo.com

Received January 2010; revised July 2010

ABSTRACT. *Conventional visual cryptography methods divide a secret digital image into n pieces and distribute them to n participants. This paper proposes a novel approach to visual cryptography for binary images that includes the capabilities of watermarking and verification. The proposed method allows an $n \times n$ watermark image to be embedded into an $n \times n$ secret image to construct two shadows and then to be used to verify the accuracy of the reconstructed image. Checking to determine the reliability of all shadows before they are used to recover the secret image prevents a participant from incidentally or deliberately providing invalid data. Moreover, our proposed method has low computation requirements, so it is suitable for real-time applications.*

Keywords: watermarking, visual cryptography, torus automorphism, authentication

1. **Introduction.** Visual cryptography (VC), also called visual secret sharing (VSS), is a technique in which a secret is encrypted into several share images and then decrypted later using a human visual system to stack all the share images [1]. In 1995, Naor and Shamir [2] were the first to propose a (t, n) visual cryptography technique that involves dividing an image into n different shares. The secret image could be reconstructed if at least t shares were stacked, where $n \geq t$. If fewer shares were available, the secret could not be decrypted [3]. Many VC methods have been proposed in the past few years; some have been designed to allow only a particular sub-group of defined participants to decode the secret [4], but some have been designed for any sub-group of k participants, and these are the most popular [5, 7]. Among them, the $(2, n)$ schemes and the (n, n) schemes get more attention.

One of the challenges in the development of VC was limiting the computational costs required in order to extract the secret. If light computational costs are involved in a VC-based application, VC will obtain more functionality and security. Therefore, VC-based applications [12, 13] have been proposed that require only light computation to disclose the secret in the decoding phase. In [12], Lukac and Plataniotis proposed bit-level-based image secret-sharing wherein a gray or color image is decomposed into several binary bit-planes. Each bit-plane is treated as a binary secret image that is processed using VC encryption to form two share images. After that, two gray or color-share images are generated by composing all the binary share images through stacking these bit-planes. Therefore, a good reconstruction is achieved by performing decryption by means of simple Boolean operation in the decomposed bit-planes.

Tsai et al. proposed an image-sharing scheme [13] that combined image-hiding and VC. In this scheme, secrets are divided into multiple parts that are hidden in the bit-planes of a set of cover images to form stego-images. The aim of this scheme is to prevent anyone who processes only one stego-image from gaining information about the secret.

As VC and VC-based applications continue to be studied extensively, the cheating problem—stacking fake and genuine shares together to reveal the secret—that exists in VC has been highlighted, and preventing cheating has become an important issue [8, 9, 10]. Horng et al. [8] proposed two methods to prevent cheating in VC. One method was an intuitive extension from the t -out-of- n scheme to the t -out-of- $(n + M)$, where the extra M transparencies must be kept secret by the dealer. The method makes it harder for the cheater to predict the structure of other transparencies. The second method was an authentication method that works by generating several shares, called authentication shares [11], that authenticate the integrity of shares prior to stacking them. Cheating is detected when the fake share is stacked with the specified authentication share. Still, both methods have drawbacks: the cheating simulation results of the $(t, n+L)$ scheme show that cheating prevention does not work well, and the second method, which requires the use of authentication shares, entails substantial additional costs to maintain the authentication shares. Hu and Tzeng [9] also claimed that there is a way to cheat in the methods proposed by Horng et al. [8] and proposed a genetic transformation from traditional VC schemes to cheat-preventing VC schemes. However, the assumption of Hu and Tzeng's scheme, that a $(2, 2)$ VC also suffers the cheating problem, is not always true because one share-holder does not know the disclosed secret prior to stacking the two shares, so the cheating results in nonsense.

Tsai et al. [10] improved Horng et al.'s authentication-based scheme [8] using a genetic algorithm that adopted multiple, distinct secret images so that each set of share pairs reveals a different secret image.

However, the cheating problem prevention schemes do not determine whether cheating attacks have happened. Instead, the participants face the recovered vague secret. Moreover, the existing cheating-prevention schemes [8, 9, 10] inherited the drawbacks of traditional VC, that is, alignment difficulties and extra cost of managing shares. To solve this problem, Zhao et al. [15] and Chang et al. [16] separately proposed verifiable secret-sharing schemes. However, most verifiable secret-sharing schemes allow participants to verify only their received shadows instead of the reconstructed secret image [15]. In 2007, Wang et al. [14] proposed two VSS schemes, one a probabilistic $(2, n)$ scheme for binary images and the other an (n, n) scheme for binary and grayscale images, in which Boolean AND and XOR operations are used. However, for the $(2, n)$ scheme, a secret image cannot be exactly reconstructed and directly extended to a (k, n) scheme.

There are four general criteria for evaluating secret sharing techniques: security, accuracy, computational complexity, and shadow size, also called pixel expansion. Designing

a scheme that meets all four criteria is the goal of many scholars. In particular, a VC scheme must satisfy the security criterion and reconstruct the secret image accurately. In this paper, we propose a new visual secret-sharing method for binary images that allows participants to verify the reconstructed secret image. To enhance participants' reliability in the image, a binary watermark image of the same size as the secret image is first embedded into generated shadows during the share construction procedure. This watermark helps participants identify whether the collected shadows were damaged. The sums and modular operation are used to generate a shadow set and a reconstructed secret image so that the scheme maintains a low computation cost, and a torus automorphism [17] is used to enhance its security. Experimental results show that our scheme satisfies the four basic criteria of a VSS scheme—security, accuracy, computational complexity and the size of shadow—and allows participants to verify their reconstructed secret images.

The rest of this paper is organized as follows. Section 2 briefly describes Wang et al.'s method. Then a detailed description of our proposed scheme is given in Section 3. Section 4 illustrates the experimental results for our proposed scheme and provides an analysis of performance on binary images. Finally, our conclusions are summarized in Section 5.

2. Preliminary. To enhance the security of the proposed scheme, after generating two shadows, we exploit a transformation named torus automorphism to permute them to be random images, sized HW . Therefore, herein, the torus automorphism is briefly introduced. The detailed of the torus automorphism is represented as below. Suppose that we have an image O sized N , the Arnold's cat map permute the pixel located at the position (x, y) to the new position (x', y') by the following formula: $\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ z & z+1 \end{bmatrix} \times \begin{bmatrix} x \\ y \end{bmatrix} \bmod N$ where \bmod is the modulo operation and $0 \leq x, y, x', y' \leq N-1$. It can be seen that the new position (x', y') is determined if the parameter z is known. Therefore, the parameter z can be kept by participants as a secret key. After some iterations of the transformation, the image O becomes a random image R . However, if we rotate further iterations of the transformation, the image R will be returned to the original image O .

3. The proposed scheme. The proposed scheme uses a watermark image to verify the reconstructed secret image so that a defined participant does not need to execute shadow verification during both the shares reconstruction phase and the revealing phase. Our scheme can be divided into two procedures: (1) the shares construction procedures, and (2) the revealing and verifying procedures. First, during the shares construction procedure, the dealer generates two shadows, called SA and SB , from the secret image I and a binary watermark L . Second, the dealer applies a torus automorphism to permute two generated shadows. During the revealing and verifying procedure, participants re-permute two collected shadows and later reconstruct a secret image I' and an extract watermark L' . A flowchart of our proposed scheme is shown in Figure 1, and detailed descriptions of two procedures are given in the following subsections.

3.1. Share construction procedure. The shares construction procedure consists of five steps, as illustrated in the flowchart in Figure 2.

Shares construction algorithm

Input : An $H \times W$ original secret image $I = (I_{ij})$, where $i = 0, 1, \dots, H-1$ and $j = 0, 1, \dots, W-1$, an $H \times W$ watermark image $L = (L_{ij})$, where $i = 0, 1, \dots, H-1$ and $j = 0, 1, \dots, W-1$.

Output: Two $H \times W$ noise-like shadow images $S^A = (S_{ij}^A)$ and $S^B = (S_{ij}^B)$, where $i = 0, 1, \dots, H-1$ and $j = 0, 1, \dots, W-1$.

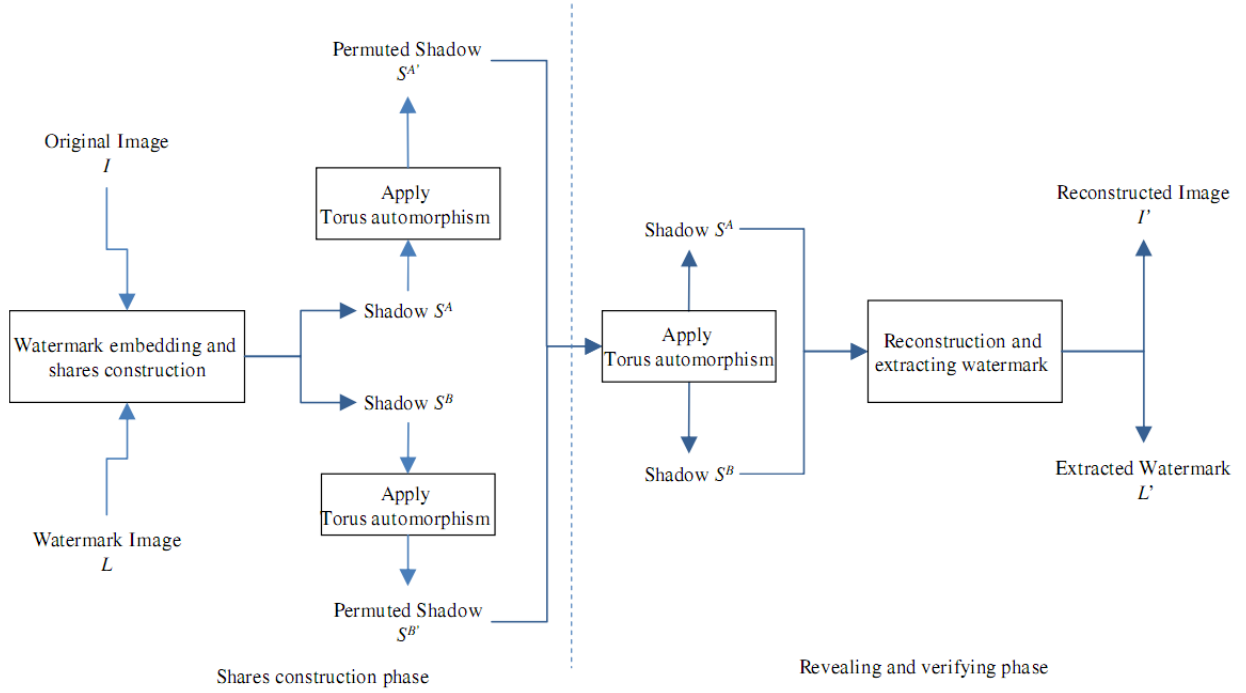


FIGURE 1. Flowchart of the proposed scheme

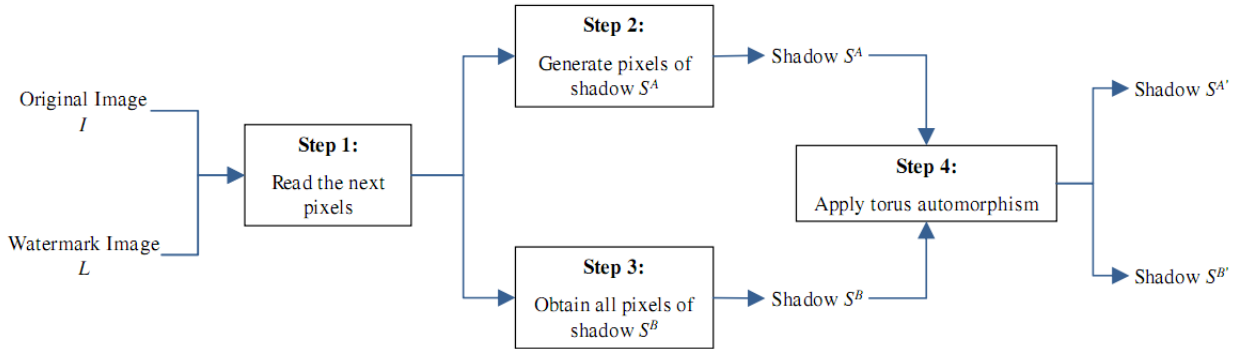


FIGURE 2. Flowchart of the share construction procedure

Step 1: Set $i = 0$ and $j = 0$, which means that the first pixels of both the secret image I and the watermark image L are considered. Read pixel I_{ij} from the secret image I and read watermark pixel L_{ij} from the watermark image L .

Step 2: Obtain pixel of shadow S^A by computing

$$S_{ij}^A = \lfloor ((I_{ij} \times 2 + L_{ij} + 1) \bmod 4) / 2 \rfloor \quad (1)$$

Step 3: Find the value for pixel of shadow S^B , using the formula:

$$S_{ij}^B = (I_{ij} \times 2 + L_{ij} + 1) \bmod 2 \quad (2)$$

Repeat Steps 2 and 3 until all pixels are processed. The outputs of this algorithm are two noise-like shadows, S^A and S^B .

Step 4: After constructing two shadows, apply a torus automorphism to permute the position of pixels in these shadows. The purpose of this step is to enhance the security of our method.

The example given here demonstrates the proposed shares construction phase.

Example 3.1. Assume that we have a 2×2 secret image I and a 2×2 watermark image L , as shown in Figures 3 and 4.

First, let us consider the first pixel of the secret image and the watermark image as $I_{1,1}$ and $L_{1,1}$, respectively. The first pixel of shadow S^A is obtained by formula (3): $S_{1,1}^A = \lfloor ((I_{1,1} \times 2 + L_{1,1} + 1) \bmod 4) / 2 \rfloor = 1$. The first pixel of shadow S^B is computed by formula (4): $S_{1,1}^B = (I_{1,1} \times 2 + L_{1,1} + 1) \bmod 2 = 0$. Similarly, we generate all the pixels of the two shadows, as shown in Figures 5 and 6.

After two shadows are generated, we permute two shadows using torus automorphism. When this step is finished, these permuted shadows will be kept private by two defined holders.

0	1
1	0

FIGURE 3. Original Secret Image I

1	1
0	1

FIGURE 4. Watermark Image L

1	1
0	1

FIGURE 5. Shadow Image S^A

0	0
1	0

FIGURE 6. Shadow Image S^B

3.2. Revealing and verifying procedure. The proposed revealing and verifying procedure allows the defined participants to extract the embedded watermark and reconstruct the secret image from the two collected shadows, S^A and S^B . The image quality of the reconstructed secret image is the same as that of the original secret image and has no distortion. Moreover, the procedure helps legitimate participants verify the reconstructed secret image I' based on the extracted watermark L' . Figure 7 is a flowchart of our proposed revealing and verifying procedure.

Revealing algorithm

Step 1: Apply the torus automorphism algorithm to shadows $S^{A'}$ and $S^{B'}$ to obtain the original shadows, S^A and S^B .

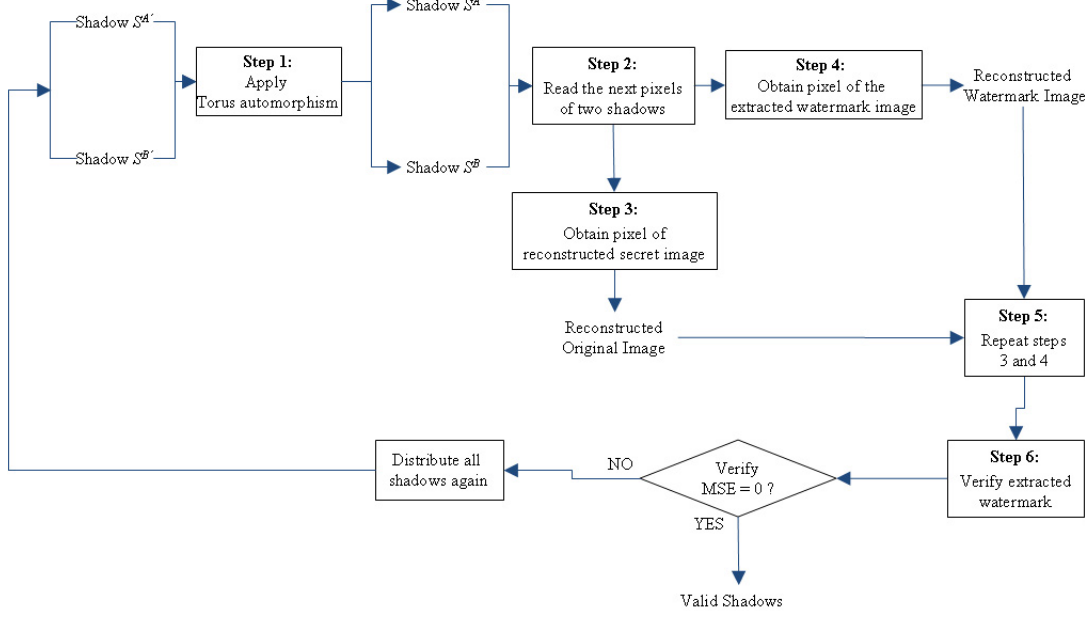


FIGURE 7. Flowchart of the share construction procedure

Step 2: Read the pixel S_{ij}^A of shadow S^A and the pixel S_{ij}^B of shadow S^B .

Step 3: Reconstruct the pixel of the reconstructed secret image using the formula:

$$I'_{ij} = \lfloor ((S_{ij}^A \times 2 + S_{ij}^B + 3) \bmod 4) / 2 \rfloor \quad (3)$$

Step 4: Obtain the pixel L'_{ij} using the formula:

$$L'_{ij} = (S_{ij}^A \times 2 + S_{ij}^B + 3) \bmod 2 \quad (4)$$

Step 5: Repeat steps 3 and 4 until all the pixels of two shadows are processed.

Step 6: Determine any difference between the original watermark and the reconstructed watermark. The extracted watermark can be verified by using mean square error (MSE). If the MSE value is equal to 0, the extracted watermark is the same as the hidden one and there has been no cheating by any participant. Otherwise, the receiver can ask the dealer to distribute all the shadows again.

The following example is given to demonstrate the revealing phase more clearly. Assume that, after applying torus automorphism to the collected shadows, we receive two 2×2 shadow images as shown in Figures 8 and 9.

1	1
0	1

FIGURE 8. Shadow Image S^A

0	0
1	0

FIGURE 9. Shadow Image S^B

The first pixel of the reconstructed secret image is computed by using formula (5): $I_{1,1} = \lfloor ((S_{1,1}^A \times 2 + S_{1,1}^B + 3) \bmod 4) / 2 \rfloor = 0$. The first pixel of shadow S^B is computed by

using formula (4): $L'_{1,1} = (S_{1,1}^A \times 2 + S_{1,1}^B + 3) \bmod 2 = 1$. Similarly, we generate all the pixels of the reconstructed image and the extracted watermark image as shown in Figures 10 and 11.

0	1
1	0

FIGURE 10. Reconstructed Secret Image I

1	1
0	1

FIGURE 11. Extracted Watermark Image L

It is clear that the reconstructed secret image I' and the extracted watermark L' shown in Figures 10 and 11 are exactly like those shown in Figures 3 and 4, so our proposed scheme can generate the same secret image I and watermark image L , without any distortion, when no cheating occurs on either shadow S^A or shadow S^B .

4. Experimental results. In this paper, we propose a (2, 2) secret-sharing scheme that satisfies the four general criteria of security, accuracy, computational complexity and shadow size, but that also archives the reversibility of the extracted secret image. To demonstrate how the scheme successfully achieves these objectives, this section produces a set of experimental results. Experimental results and discussions about the four general criteria are presented in subsections 4.1, 4.2 4.3 and 4.4. To illustrate the features of our proposed scheme, the comparisons between our scheme, Wang et al.'s scheme [30] and Lukac and Plataniotis' scheme are presented in subsection 4.5.

To illustrate our scheme, we used a set of test images shown in Figure 12. The set contains six 512x512 binary images: "Lena," "Peppers," "Baboon," "F16," "GoldHill," and "Boat." In the test sets, "Lena," "Peppers" and "Baboon" serve as the secret image, while "Boat," "GoldHill" and "F16" serve as the watermarks that are used to verify the reconstructed secret images.

4.1. Security analysis. To guarantee that our scheme satisfies the security criterion in that it avoids leaking any information about the original secret image, torus automorphism is used to relocate the constructed shadows after they have been generated. The sets of shadows generated by our scheme are shown in Figures 13.

4.2. Accuracy. In our experiments, the peak signal-to-noise ratio ($PSNR$) defined in Equation (7) is used to evaluate the quality of the reconstructed binary images. In general, a higher $PSNR$ means that the quality of the reconstructed image is better. Basically, $PSNR$ value should range from 30dB to 40dB if a scheme offers good visual quality.

$$PSNR = 10 \times \log_{10} \frac{255^2}{MSE} \quad (5)$$

where MSE is the mean square error between the original image and reconstructed image. For an original grayscale image with a size of $H \times W$, the corresponding MSE is defined in Equation (8).

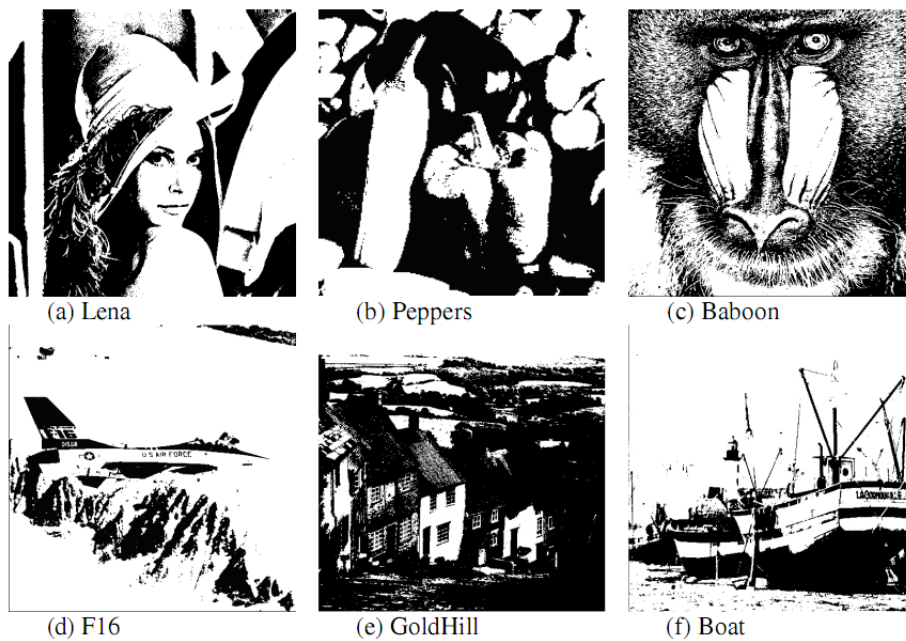


FIGURE 12. Six binary test images

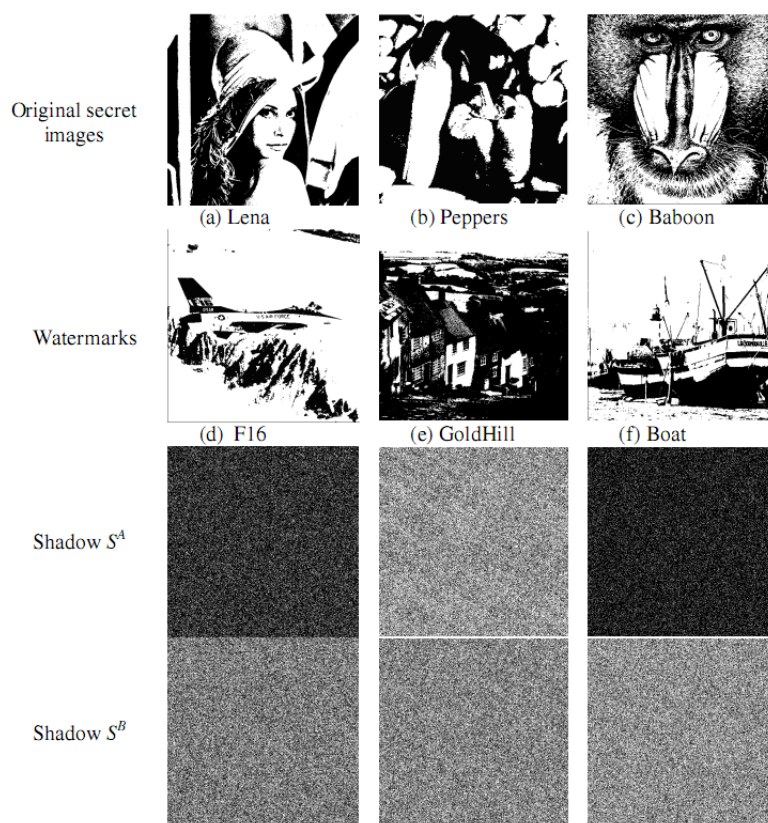


FIGURE 13. Shadow images generated by our proposed scheme

$$MSE_Q = \frac{1}{W \times H} \sum_{x=1}^W \sum_{y=1}^H (Q_{xy} - Q'_{xy})^2 \quad (6)$$

where Q_{xy} and Q'_{xy} are the pixel values at position (x, y) of the original secret image and the reconstructed secret image, respectively.

Our proposed scheme can reconstruct the secret image with no distortion. Therefore, the expected MSE value is equal to 0. To determine the accuracy of our proposed scheme, three original secret images and their reconstructed secret images are shown in Figure 14.






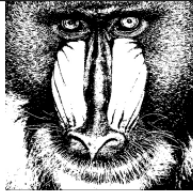

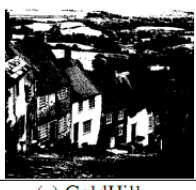




Original secret images				
	(a) Lena	(b) Peppers	(c) Baboon	
	Reconstructed secret images			
Original watermarks				
		(d) F16	(e) GoldHill	(f) Boat
	Extracted watermarks			
Similarity of original secret image using our scheme		100% (exactly the same as the original secret image)		
Similarity of original watermark using our scheme		100% (exactly the same as the original watermark image)		

FIGURE 14. Quality of reconstructed images with binary test images

4.3. Computational complexity. According to the descriptions of our proposed scheme, the computational cost depends on only two operations: the sums operation and torus automorphism. Clearly, the complexity of the sums operation is very low and has very little effect on the computational complexity of our scheme. The experimental results, including the execution time of our scheme, are shown in Table 1.

TABLE 1. Computational complexity of our scheme with binary images

Binary images		
Lena	Peppers	Baboon
0.547s	0.546s	0.547s

4.4. Pixel expansion. During the share construction process, our proposed scheme generates two shadows that have the same size of the original secret image. As formulas (3) and (4) show, each pixel L_{ij} and I_{ij} located at the i th row and the j th column of the watermark image and the secret image, respectively, is used to generate the (i, j) pixel of shadows. Thus, our proposed scheme does not cause a pixel expansion problem. Table 2 shows the correlation between the original image size and the shadow size.

TABLE 2. Correlation between original image size and its shadow size

Image	Binary images
Size of the original image	262144bytes
Size of the shadow S^A	262,144bytes
Size of the shadow S^B	262,144bytes

4.5. Evaluation of verifying. The mean square error (MSE) in Equations (7) and (8) is used to identify whether cheating has occurred by comparing the content of an original watermark with that of an extracted one. If the MSE value is equal to zero, there is no difference between the two watermarks. If the MSE value is not equal to zero, the reconstructed secret image I' is unreliable.

Figure 15 shows the extracted watermarks, the reconstructed secret images and their verification results. In this scenario, one shadow is partially damaged; the corresponding MSE of the extracted watermark is not equal to “0” for “Peppers” and “Baboon” because the shadow S^A of “Peppers” and the shadow S^B of “Baboon” are damaged. The two shadows of “Lena” in Figure 15(a) are unchanged.

4.6. Comparisons. To demonstrate the features of our proposed scheme, this subsection compares our scheme with Wang et al.’s scheme [30] and Lukac and Plataniotis’s scheme [25] in terms of number of shadows, the style of the secret image, shadow size, computational complexity, MSE value, and verifying ability. - *Number of shadows:* Both our proposed scheme and Wang et al.’s scheme can be used to generate two shadows during shares construction phase while Lukac and Plataniotis’ scheme can generate n ($n > 2$) shadows.

- *Style of secret image:* Style of secret image is used to specify that the secret image is binary, grayscale or color image. In this term, the binary images can be used in our scheme and Wang et al.’s scheme. In, Lukac and Plataniotis’s scheme, binary image is also adopted to generate shadows. However, their scheme can be extended for grayscale images and color images. - *Shadow size:* This criterion is considered to determine whether pixel expansion problem occurs. Even for binary, grayscale or color images, Lukac and Plataniotis’s scheme generates a set of shadows whose size is twice bigger than that of the original secret image. On the contrary, the size of shadows generated by our scheme and Wang et al.’s scheme is equal to the original secret one.

- *Verifying ability:* We take this criterion to evaluate whether the scheme can be applied to verify the reconstructed image. It is clear that only our scheme can verify the reconstructed images to make sure that the collected shadows are valid; while Wang et al.’s scheme and Lukac and Plataniotis’s scheme cannot verify the image after reconstructing.

- *Computational complexity:* Wang et al.’s scheme uses Boolean operations, AND and XOR. All of them are low complex operations. However, in the construction phase, it needs more time to generate $(n + 1)$ random matrices and then compute n intermediate matrices before constructing shadows. In Lukac and Plataniotis’s scheme, the halftoning




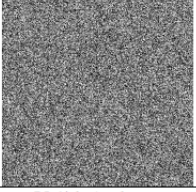
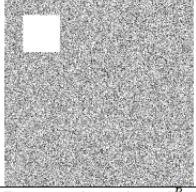
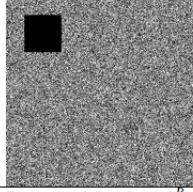






Original watermarks			
Shadow is partially damaged			
	Valid shadow	Damaged shadow S^B	Damaged shadow S^B
Reconstructed Watermarks			
	Exactly the same	PSNR = 12	PSNR = 10
Reconstructed secret image			
	Exactly the same	Similar	Similar
Reliability	$MSE = 0$ Sure	$MSE \neq 0$ Not sure	$MSE \neq 0$ Not sure

FIGURE 15. Quality of reconstructed images when one shadow is partially damaged

and inverse halftoning steps are required during encryption and decryption phases, respectively. Our proposed scheme only uses the basic operations and after that employs torus automorphism to reorganize the shadows. Thus, the computational cost is very low. In other words, the computational complexity of our scheme is lower than others.

- *MSE value*: *MSE* is used to evaluate the different between extracted watermark and the original watermark. If the *MSE* is equal to zero, it is said that the extracted watermark is exactly recovered, and the reconstructed secret image can be trusted. Otherwise, the extracted watermark is unreliable. In our scheme, both the extracted watermark and the secret image are the same as the original ones. Lukac and Plataniotis's scheme also can reconstruct the original secret image completely. In contrast, the reconstructed secret image with Wang et al. is not the same as the original one.

- *Style of shadow*: This term is discussed to determine that the shadows are noise-like or meaningful shadows. In this criterion, all of the shadows generated by our scheme, Wang et al.'s scheme and Lukac and Plataniotis's scheme are noise-like shadows.

The abridgment of these criteria is shown in Table 3.

5. Conclusions. The paper proposes a visual secret-sharing scheme for binary images. Our proposed scheme satisfies the four general criteria required for visual secret sharing systems and also offers verifiability not available in previous visual secret-sharing scheme. Moreover, the computational complexity of our proposed scheme is very low, so it is suitable for real-time applications.

TABLE 3. Comparison between our scheme, Wang et al.'s scheme and Lukac and Plataniotis' scheme

Criteria	Wang et al.'s scheme [30]	Lukac and Plataniotis' scheme [25]	Proposed scheme
Number of shadows (n)	$n \geq 2$	$n \geq 2$	$n = 2$
Style of secret image	Binary, or grayscale	Binary, grayscale or color	Binary
Verifying ability	No	No	Yes
Shadow size	N	$2N$	N
Computational complexity	Low	Low	Lower
MSE value	$MSE > 0$	$MSE = 0$	$MSE = 0$
Style of shadow	Noise-like shadows	Noise-like shadows	Noise-like

Acknowledgment. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

- [1] A. Shamir, How to share a secret, *Communications of the Association for Computing Machinery*, vol. 22, no. 11, pp. 612-613, 1979.
- [2] G. R. Blakley, Safeguarding cryptographic keys, *Proc. of National Computer Conference*, American Federation of Information Processing Societies, pp. 313-317, 1979.
- [3] M. Naor and A. Shamir, Visual cryptography, *Lecture Notes Computer Science*, vol. 50, pp. 1-12, 1995.
- [4] G. J. Simmons, An introduction to shared secret and/or shared control schemes and their application, *Contemporary Cryptology, The Science of Information Integrity*, IEEE Press, New York, pp. 441-497, 1992.
- [5] D. R. Stinson, An explication of secret sharing schemes, *Designs, Codes and Cryptography*, vol. 2, pp. 35-390, 1992.
- [6] G. Ateniese, C. Blundo, A. D. Santis, and D. R. Stinson, Visual cryptography for general access structures, *Information Computation*, vol. 129, no. 2, pp. 86-106, 1996.
- [7] G. Ateniese, C. Blundo, A. De Santis, and D. R. Stinson, Extended capabilities for visual cryptography, *Theoretical Computer Science*, vol.250, pp.143-161, 2001.
- [8] C. Thien, and J. C. Lin, Secret Image Sharing, *Computers and Graphics*, vol. 26, no. 1, pp. 765-770, 2002.
- [9] C. C. Thien, and J. C. Lin, An image-sharing method with user-friendly shadow images, *IEEE Trans. Circuits and System for Video Technology*, vol. 13, no. 12, pp. 1161-1169, 2003.
- [10] R. Z. Wang, and C. H. Su, Secret image sharing with smaller shadow images, *Pattern Recognition Letter*, vol. 27, no. 6, pp. 551-555, 2006.
- [11] Y. S. Wu, C. C. Thien, and J. C. Lin, Sharing and hiding secret images with size constraint, *Pattern Recognition*, vol. 37, no. 7, pp. 1377-1385, 2004.
- [12] C. C. Chang, and I. C. Lin, A new (t, n) threshold image hiding scheme for sharing a secret color image, *Proc. of ICCT*, Beijing, China, vol. 1, pp. 196-202, 2003.

- [13] J. B. Feng, H. C. Wu, C. S. Tsai, and Y. P. Chu, A new multi-secret images sharing scheme using Lagrange's interpolation, *Journal of Systems and Software*, vol. 76, no. 3, pp. 327-339, 2005.
- [14] T. Hofmeister, M. Krause, and H. U. Simon, Contrast-optimal k out of n secret sharing schemes in visual cryptography, *Theoretical Computer Science*, vol. 240, no. 2, pp. 471-485, 2000.
- [15] C. Blundo, A. De Santis and D. R. Stinson, On the contrast in visual cryptography schemes, *Journal of Cryptology*, vol. 12, no. 4, pp. 261-289, 1999.
- [16] C. N. Yang, and C. S. Lai, New colored visual secret sharing schemes, *Designs, Codes and Cryptography*, vol. 20, no. 3, pp. 325-335, 2000.
- [17] S. Cimato, R. De Prisco, and A. De Santis, Contrast optimal colored visual cryptography schemes, *Proc. of IEEE Information Theory Workshop*, Paris, France, pp. 139-142, 2003.
- [18] C. Blundo, A. De Santis, and M. Naor, Visual cryptography for gray level images, *Information Processing Letters*, vol. 75, no. 6, pp. 255-259, 2000.
- [19] R. Ito, H. Kuwakado, and H. Tanaka, Image size invariant visual cryptography, *IEICE Trans. Fundamentals*, vol. E82-A, no. 10, pp. 2172-2177, 1999.
- [20] G. Horng, T. H. Chen, and D. S. Tsai, Cheating in visual cryptography, *Designs, Codes, Cryptography*, vol. 38, pp. 219-236, 2006.
- [21] C. M. Hu, and W. G. Tzeng, Cheating prevention in visual cryptography, *IEEE Trans. Image Processing*, vol. 16, no. 1, pp. 36-45, 2007.
- [22] D. S. Tsai, T. H. Chen, and G. Horng, A cheating prevention scheme for binary visual cryptography with Homogeneous secret image, *Pattern Recognition*, vol. 40, no. 8, pp. 2356-2366, 2007.
- [23] D. S. Tsai, and G. Horng, Cheating in visual cryptography revisited, *Proc. of the 17th Information Security Conference*, Chiayi, Taiwan, 2007.
- [24] M. Naor, and B. Pinkas, Visual authentication and identification, *Proc. of advances in Cryptology*, Lecture Notes in Computer Science, vol. 1294, pp. 322-336, 1997.
- [25] R. Lukac, and K. N. Plataniotis, Bit-level based secret sharing for image encryption, *Pattern Recognition*, vol. 38, no. 5, pp. 767-772, 2005.
- [26] C. S. Tsai, C. C. Chang, and T. S. Chen, Sharing multiple secrets in digital images, *The Journal of Systems and Software*, vol. 64, pp. 163-170, 2002.
- [27] C. C. Chang, and J. C. Chang, An image intellectual property protection scheme for gray-level images using visual secret sharing strategy, *Pattern Recognition Letter*, vol. 23, pp. 931-941, 2002.
- [28] S. J. Lin, and J. C. Lin, VCPSS: A two-in-one two decoding-options image sharing method combining visual cryptography and polynomial-style sharing (PSS) approaches, *Pattern Recognition*, vol. 40, no. 12, pp. 3652-3666, 2007.
- [29] D. Jin, W. Q. Yan, and M. S. Kankanhalli, Progressive color visual cryptography, *Journal of Electronic Imaging*, vol. 14, 2005.
- [30] D. Wang, L. Zhang, N. Ma, and X. Li, Two secret sharing schemes based on boolean operations, *Pattern Recognition*, vol. 40, pp. 2776-2785, 2007.
- [31] R. D. Prisco, and A. D. Santis, Cheating immune (2, n)-threshold visual secret sharing, *Proc. of Security and Cryptography for Networks*, vol. 4116, pp. 216-228, 2006.
- [32] C. C. Lin, and W. H. Tsai, Secret image sharing with steganography and authentication, *Journal of Systems and Software*, vol. 73, no. 3, pp. 405-414, 2004.
- [33] C. N. Yang, T. S. Chen, K. H. Yu, and C. C. Wang, Improvements of image sharing with steganography and authentication, *Journal of Systems and Software*, vol. 80, no. 7, pp. 1070-1076, 2007.
- [34] R. Zhao, J. J. Zhao, F. Dai, and F. Q. Zhao, A new image secret sharing scheme to identify cheaters, *Computer Standards and Interfaces*, vol. 31, no. 1, pp. 252-257, 2009.
- [35] C. C. Chang, Y. P. Hsieh, and C. H. Lin: Sharing secrets in stego images with authentication, *Pattern Recognition*, vol. 41, no. 10, pp. 3130-3137, 2008.
- [36] G. Voyatzis, and I. Pitas, Applications of toral automorphisms in image watermarking, *Proc. of IEEE International Conference on Image Processing*, Lausanne, Switzerland, vol. 2, pp. 237-240, 1996.
- [37] G. Matthew, K. Bruce, and Z. George, Visualizing toral automorphisms, *The Mathematical Intelligencer*, vol. 15, no. 1, pp. 63-66, 2003.