Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE @ DIRECT®

Pattern Recognition Letters 24 (2003) 349–358

Pattern Recognition Letters

www.elsevier.com/locate/patrec

Visual cryptography for gray-level images by dithering techniques

Chang-Chou Lin, Wen-Hsiang Tsai [*]

Department of Computer and Information Science, National Chiao Tung University, 1001 Ta Hsueh Road, 30050 Hsinchu, Taiwan, ROC

Received 10 August 2001; received in revised form 2 July 2002

Abstract

A (k, n)-threshold visual cryptography scheme is proposed to encode a secret image into n shadow images, where any k or more of them can visually recover the secret image, but any $k-1$ or fewer of them gain no information about it. The decoding process of a visual cryptography scheme, which differs from traditional secret sharing, does not need complicated cryptographic mechanisms and computations. Instead, it can be decoded directly by the human visual system. Previous efforts in this topic are almost restricted in processing binary images, which are insufficient for many applications. In this paper, a new visual cryptography scheme suitable for gray-level images is proposed. Instead of using gray subpixels directly to construct shares, a dithering technique is used first to convert a gray-level image into an approximate binary image. Then existing visual cryptography schemes for binary images are applied to accomplish the work of creating shares. The overall effect of the proposed method is the achievement of visual encryption and decryption functions for gray-level images. Some comparisons with a previously proposed method are also made. Some experimental results are shown to prove the feasibility of the proposed method. Finally, an application is mentioned to show its practicability.

Keywords: Visual cryptography; Secret sharing; Watermarking; Dithering

1. Introduction

How to keep a secret is always an important issue in many applications. Two major approaches to this aim are information hiding and secret sharing. A common method for information hiding is to use the watermarking technique (Cox et al., 2001; Katzenbeisser and Petitcolas, 2000; Johnson et al., 2000). And a well-known technique for secret sharing is the cryptography method proposed by Shamir (1979). For sharing images, Naor and Shamir proposed further the idea of visual cryptography (Naor and Shamir, 1995) in 1994. Following their work, several extensions have been proposed (Droste, 1996; Hofmeister et al., 1997; De Bonis and De Santis, 2000). A (k, n)-threshold visual cryptography scheme is a

[*] Corresponding author. Tel.: +886-35-712121/720631; fax: +886-35-721490/727382.

E-mail address: whtsai@cis.nctu.edu.tw (W.-H. Tsai).

0167-8655/03/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
PII: S0167-8655(02)00259-3

method to encode a secret image into n shadow images called shares, where any k or more of them can be combined visually to recover the secret image, but any $k-1$ or fewer of them gain no information about it. The visual recovery process consists of xeroxing the shares onto transparencies, and then stacking them to obtain a "decoded" image visually approximating the original secret image. This basic model has been applied to many applications, which include information hiding (Naor and Shamir, 1995), general access structures (Ateniese et al., 1996a,b), visual authentication and identification (Naor and Pinkas, 1997), and so on. Unfortunately, these applications are all restricted to the use of binary images as input due to the nature of the model. This drastically decreases the applicability of visual cryptography because binary images are usually restricted to represent text-like messages. The characteristics of images for showing object shapes, positions, and brightness thus cannot be utilized. At the age of Internet, data of image form gradually replace data of text form. It is not sufficient that visual cryptography schemes can only deal with binary images. Verheul and van Tilborg (1997) first tried to extend visual cryptography into gray-level images. The details of their scheme will be described in Section 3. They used the gray levels existing in original images to form shares instead of using black and white values only. But in ordinary situations, their method has the disadvantage of size increase in the decoded image. So in this paper, we propose a new method suitable for sharing gray-level images. The method utilizes the technique of digital image halftoning first to transform a gray-level image into an approximate binary image. Then the visual cryptography scheme used for binary images is applied. Some advantages of the proposed approach are that it can inherit any developed technique for binary images and that it results in less increase of the image size.

The remainder of this paper is organized as follows. In Section 2, we briefly review the (k,n)-threshold visual cryptography scheme for binary images, which is the basis of our approach. In Section 3, we introduce the method proposed in (Verheul and van Tilborg, 1997) and analyze it. In Section 4, the details of our approach based on the image halftoning technique are described. Some experimental results are shown in Section 5. A possible application to prove the practicability of our approach is proposed in Section 6. Finally, some conclusions are given in Section 7.

2. Review of $(\boldsymbol{k},\boldsymbol{n})$-threshold visual cryptography

We review the (k,n)-threshold visual cryptography scheme proposed in (Naor and Shamir, 1995) for binary images first in this section. Before an input image is encrypted into n shares with each share being delivered to a participant in the secret-sharing process, each pixel in the input image is expanded into a group of subpixels (say, including b ones) which are then assigned proper values (0 for white and 1 for black) to yield corresponding shares. To do this, two sets of $n \times b$ Boolean matrices, say denoted by C_0 and C_1 need be selected properly and systematically. Each row in each matrix in C_0 or C_1 represents the values of a group of b subpixels in a share. To share a white pixel in the input image, one of the matrices in C_0 is randomly chosen; and to share a black pixel, one of the matrices in C_1 is randomly chosen. The n rows of each chosen matrix are distributed over the n participants, each receiving a row. Let $z(\underline{v})$ denote the number of zero-valued elements in a vector v. A pair of C_0 and C_1 is considered proper if the following three conditions are all met:

(1) The result v of applying the "OR" operation to any k of the n rows of any matrix in C_0 satisfies $z(\underline{v}) \geqslant h$, where h is a non-negative integer.
(2) The result v of applying "OR" to any k of the n rows of any matrix in C_1 satisfies $z(\underline{v}) \leqslant l$, where l is another non-negative integers *smaller than h.*
(3) For any q positive integers $i_1, i_2, \ldots, i_q$ with $q < k$ and $i_1 < i_2 < \cdots < i_q \leqslant n$, the two collections D_0 and D_1 of $q \times b$ matrices, which are obtained respectively from C_0 and C_1 by restricting each $n \times b$ matrix in C_0 and C_1 to rows $i_1, i_2, \ldots, i_q$, are *indistinguishable* in the sense that they contain the same matrices with the same frequencies.

The third condition above has the previously mentioned effect that any $k-1$ or fewer of the shares provide insufficient information for decrypting the original image. Sometimes, it is desired to know the resulting "contrast" in the decoded image, which is defined in (Naor and Shamir, 1995) to be the value $(h-l)$. The larger the value, the better the contrast.

To clarify the usage of the (k,n)-threshold visual cryptography scheme, we demonstrate the (2, 2)-threshold case by an example here. First, define two matrices:

$$A_0 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}, \quad A_1 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix};$$

and then select the two sets C_0 and C_1 mentioned above to be:

$$C_0 = \{\text{all the matrices obtained by permuting the columns of } A_0\};$$

$$C_1 = \{\text{all the matrices obtained by permuting the columns of } A_1\}.$$

It may be checked that C_0 and C_1 meet the three conditions described above. Fig. 1(a) shows the content of C_0 and Fig. 1(b) that of C_1. Doing permutations can create more row patterns, all with equal probabilities of appearance; and this effect prevents the regularity of the patterns (e.g., the pattern 0101 implies black in the original image) from being found out by an adversary. On the other hand, no matter how the columns are permuted, the result of stacking is consistent in contrast. In this example, it can be seen from the row contents of A_0 and A_1 that a decoded white pixel will be composed of two white subpixels and two black subpixels while a decoded black pixel will be composed of four black subpixels. To encrypt a white pixel in the original image, we take *share*1 to be the first row of a matrix R_0 randomly chosen from C_0 (i.e., we assign the subpixel values of *share*1 to be the content of the first row of R_0), and *share*2 to be the second row of R_0. The visual patterns of the two shares are depicted in Fig. 2(a) and (b). The result of stacking is shown in Fig. 2(c). Similarly, we choose R_1 randomly in C_1 to encrypt a black pixel, and the corresponding two shares and their stacking result are shown in Fig. 3. This completes the encryption of a pixel in the input image, and the other pixels are processed in a similar way. Fig. 4 shows additionally the result of

$$\begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

(a)

$$\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$
$$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

(b)

Fig. 1. C_0 and C_1 of an example of (2,2)-threshold visual cryptography. (a) Matrices in C_0 which are all the permutations of A_0, (b) matrices in C_1 which are all the permutations of A_1.

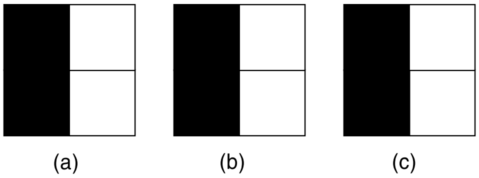


Fig. 2. The shares and decoded white pixel of the example of (2,2)-threshold visual cryptography. (a) The visual pattern of *share*1, (b) the visual pattern of *share*2, (c) decoding result of a white pixel.

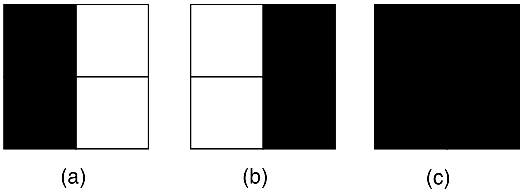


Fig. 3. The shares and decoded black pixel of the example of (2,2)-threshold visual cryptography. (a) The visual pattern of *share*1, (b) the visual pattern of *share*2, (c) decoding result of a black pixel.

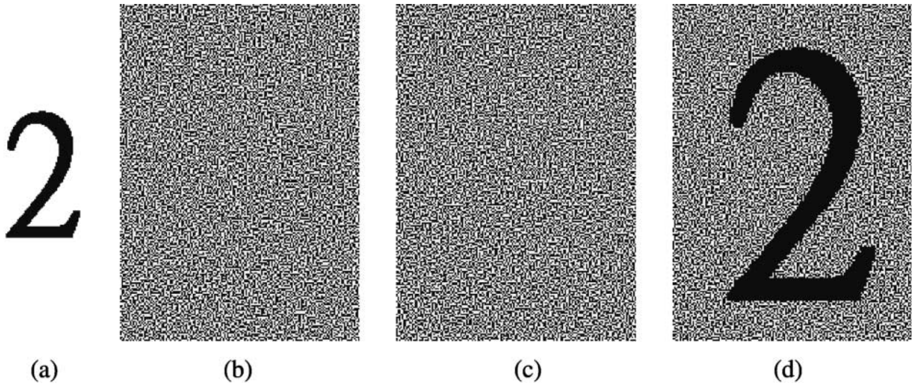


Fig. 4. An example of (2,2)-threshold visual cryptography applied to a real simple image. (a) The original image, (b) the *share*1, (c) the *share*2, (d) the decoded image.

applying the previously described (2, 2)-threshold visual cryptography scheme to an image.

The next case we want to demonstrate is an example of (2,4)-threshold visual cryptography, which is solved by choosing C_0 and C_1 respectively as:

$$C_0 = \{\text{all the matrices obtained by permuting the columns of } A_0\};$$

$$C_1 = \{\text{all the matrices obtained by permuting the columns of } A_1\}.$$

where

$$A_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

and

$$A_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

By inspecting the contents of A_0 and A_1 we see that any single share produced from a matrix R_i either in C_0 or C_1 is a random choice of a case from four ones, with each case corresponding to a row in R_i and being with one black and three white subpixels. Thus, the stacking result of any two shares of a white pixel consists of one black and three white subpixels, whereas the stacking result of any two shares of a black pixel consists of two black and two white subpixels. The difference between the two stacking results become clearer as we stack additional shares. Detailed descriptions of the process for constructing shares using more general (k, n)-threshold visual cryptography schemes can be found in (Droste, 1996; Hofmeister et al., 1997; De Bonis and De Santis, 2000; Ateniese et al., 1996a,b; Verheul and van Tilborg, 1997).

3. Survey of related works

Topics about visual cryptography for gray-level images are seldom discussed. Verheul and van Tilborg (1997) described a general method for (k, n)-threshold visual encryption of gray-level images. We review their method briefly here. For an image with c gray-levels, expand first a pixel into b subpixels. Each subpixel may take one of the gray levels of $0, 1, \ldots, c-1$. After all shares are stacked, gray level i is revealed if corresponding subpixels of all shares are of gray-level i; otherwise, the level of "black" (with the smallest gray level value) is revealed. As an illustration, we describe an example for the case of a (3,3)-threshold scheme. If there are three gray levels, we construct three collections of matrices belonging to gray levels 0, 1, 2, respectively, in the following:

C_0 = {all the matrices obtained by permuting the columns of A_0},
C_1 = {all the matrices obtained by permuting the columns of A_1},
C_2 = {all the matrices obtained by permuting the columns of A_2},

where

$$A_0 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\ 0 & 1 & 2 & 1 & 2 & 0 & 2 & 0 & 1 \end{pmatrix},$$

$$A_1 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\ 2 & 0 & 1 & 0 & 1 & 2 & 1 & 2 & 0 \end{pmatrix},$$

and

$$A_2 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\ 1 & 2 & 0 & 2 & 0 & 1 & 0 & 1 & 2 \end{pmatrix}.$$

To encrypt a pixel of gray-level 0, we take *share*1 to be the first row of a matrix R_0 randomly chosen in C_0 *share*2 the second row of R_0 and *share*3 the third row of R_0 The visual patterns of the shares and their stacking result are depicted in Fig. 5. We can see in the stacking result that the subpixel of all these three shares at the top-left corner is of gray-level 0 that is the original gray-level value of the encoded pixel, while the gray-level "black" appears in the other positions. The encryption results of the other two gray levels can be derived similarly.

It is easy to see that this scheme yields decoded images with expanded sizes. More precisely, after encrypting an image with c gray levels using the (k,n)-threshold visual cryptography scheme proposed in (Verheul and van Tilborg, 1997), the size increase as derived in (Verheul and van Tilborg, 1997) is with a factor c^{k-1} at least when $c \geqslant n$.

4. Proposed scheme

With the (k,n)-threshold visual cryptography scheme for binary images proposed in (Naor and Shamir, 1995), the theoretical increase of size is with a factor n^{k-1}, which is comparatively smaller than the scheme proposed by Verheul and van Tilborg (1997) in ordinary situations with $c \geqslant n$. Therefore, if we can convert first a gray-level image into an approximate binary image with the same size and then apply the scheme proposed in (Naor and Shamir, 1995), we can get a smaller increase in the image size than the Verheul and van Tilborg scheme. Ordered dithering is a technique satisfying this purpose. That is, it can be employed to conduct fast and parallel transformation of a gray-level image into an equal-sized binary one. For this, we adopt in this study the space-filling curve ordered dithering (SFCOD) algorithm (Zhang, 1998) that has the merit of keeping image quality by determining dither thresholds along a space-filling curve. Let I be an $m \times m$ gray image, H the corresponding halftoned binary image, $M(m,m)$ the map of a traversal-order number of the pixels of the image I along a space-filling curve over I (a simple example using Hilbert curve to divide the pixels of a 4×4 image into 16 classes is shown in Fig. 6, where the numbers in the grid are

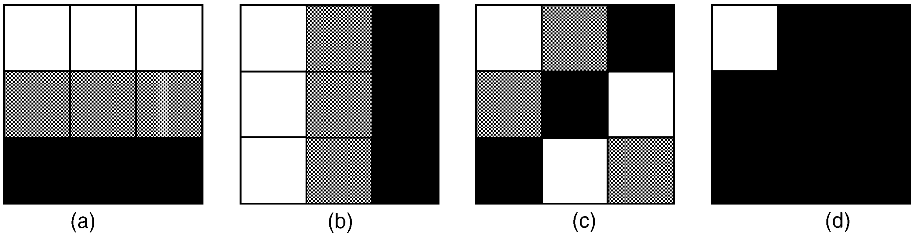


Fig. 5. The shares and decoded pixel with gray-level 0 of the example of (3, 3)-threshold visual cryptography. (a) The visual pattern of *share*1, (b) the visual pattern of *share*2, (c) the visual pattern of *share*3, (d) decoding result of a pixel with gray-level 0.

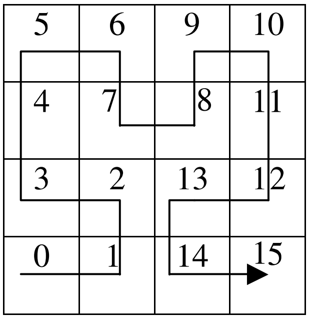


Fig. 6. Traversal order determined by a Hilbert curve.

the traversal-order numbers), and B a space-filling curve dither array (with the array contents being a permutation of $0, 1, 2, \ldots, l-1$ where l is the array length and is less than m). Then the SFCOD algorithm can be described as a procedure in the following.

```
PROCEDURE SFCOD($MB, l, I, H$)
BEGIN
FOR $i = 0$ TO $m - 1$ DO
    FOR $j = 0$ TO $m - 1$ DO
        IF
        $I(i,j)/256 \geqslant (B(M(i,j)\,\mathrm{MOD}\,l) + 0.5)/l$
        THEN $H(i,j) = 1$ ELSE $H(i,j) = 0$;
END
```

This algorithm can be regarded to divide the input image into non-overlapping blocks whose size is $\sqrt{l} \times \sqrt{l}$ and assign each pixel a value equal to the corresponding value of the traversal-order number along the space-filling curve in the block. Then the assigned value, say j, of each pixel is taken to be the index of an element of the dither array B, yielding a mapped value $B(j)$. Finally, the mapped values are used as the thresholding values to binarize the input gray-level image.

In short, we transform an input gray-level image into an approximate binary image and apply to it the visual cryptography scheme proposed in (Naor and Shamir, 1995). The overall effect is that we get a result of encrypting a gray-level image, thus achieving (k, n)-threshold visual cryptography.

5. Experimental results

In this section, three gray images are used to evaluate the performance of our proposed scheme. The reason of choosing these images is that they contain sufficient image details and gray levels. Such images are good for evaluating the effect of halftoning and visual cryptography. Fig. 7 shows an original gray image with 16 gray levels. The result of applying the SFCOD with the parameters

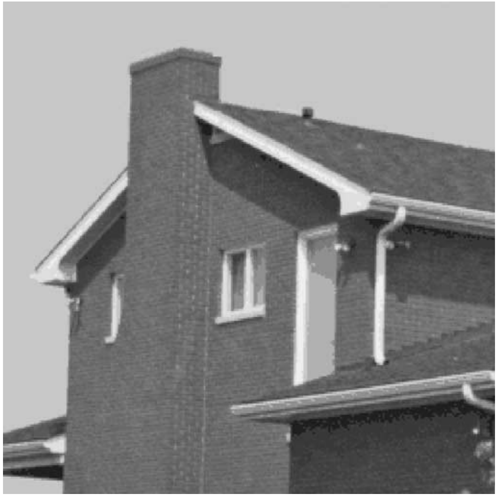


Fig. 7. The original image.

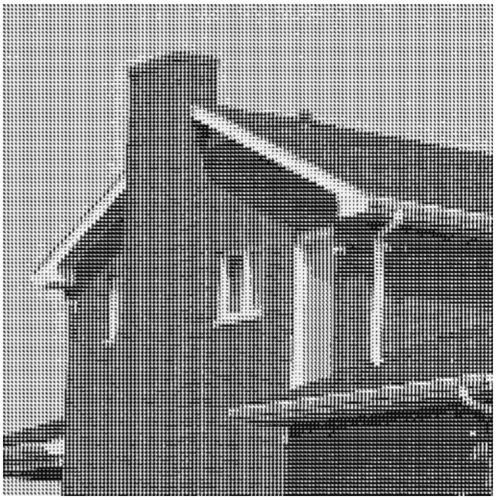


Fig. 8. The image after using SFCOD.

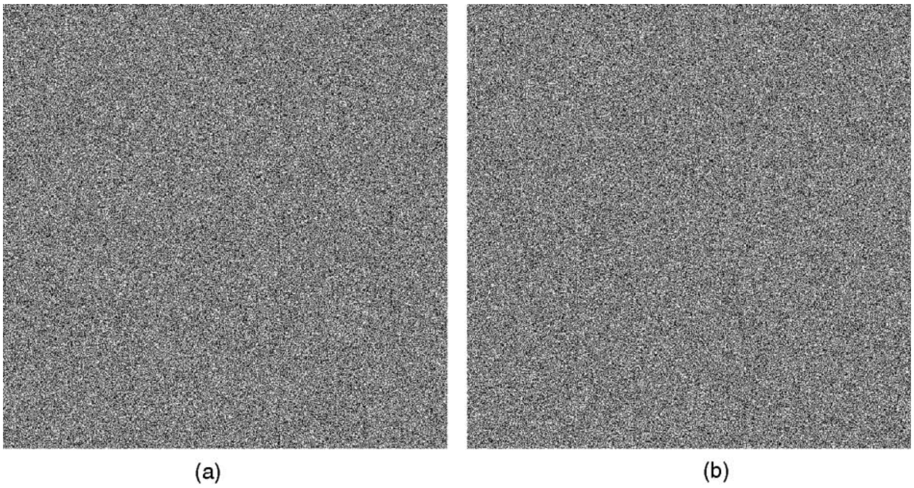


Fig. 9. The images of the shares. (a) The *share*1, (b) the *share*2.

suggested in (Zhang, 1998), where $B(i) = i$, $i = 0, 1, 2, \ldots, l - 1$, $l = 16$, and $M(i, j)$ is using the Hilbert curve to map the value of traversal-order number, is shown in Fig. 8. Then we perform the (2, 2)-threshold visual cryptography scheme. The corresponding two shares, whose sizes are both four times that of the image shown in Fig. 7, are shown in Fig. 9. The result of decoding the two shares is shown in Fig. 10, which is also with a size four times that of the image shown in Fig. 7. We observe that most details can be revealed in the image of Fig. 10 and the size of the image is just $1/4$ of the image used by the Verheul and van Tilborg scheme. Figs. 11 and 15 are two other gray images with 256 gray-levels. The results of applying the SFCOD algorithm with the same parameters as used in the first experiment are shown in Figs. 12 and 16, respectively. Their corresponding two shares are shown in Figs. 13 and 17, respectively. The results of decoding the two shares are

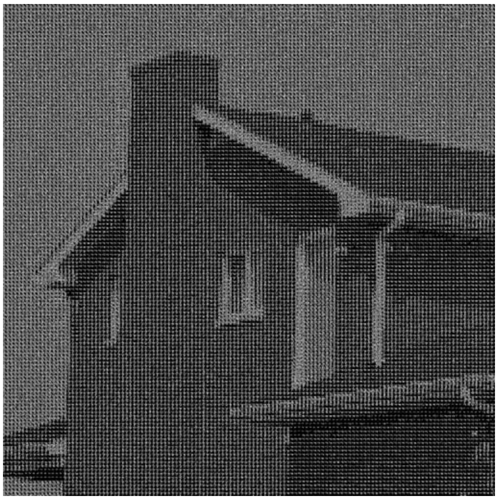


Fig. 10. The decoded image.

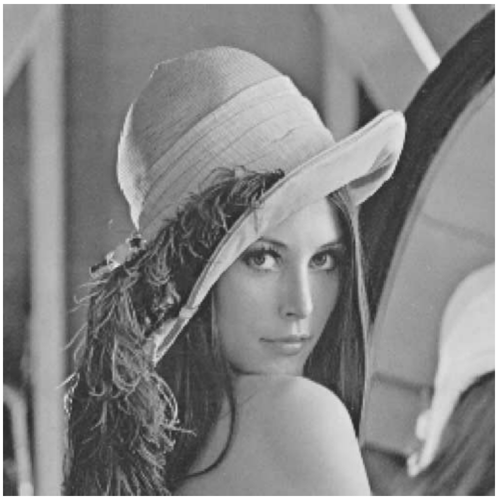


Fig. 11. The original image.

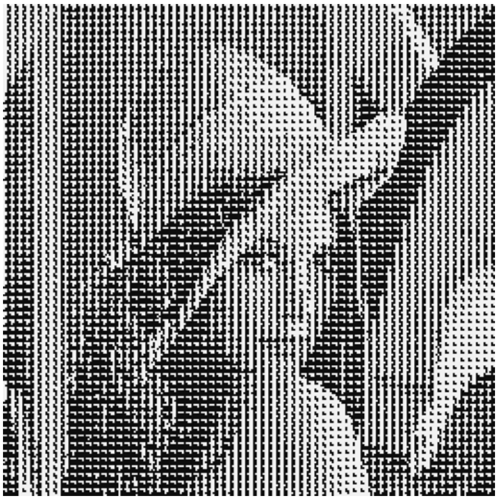


Fig. 12. The image after using SFCOD.

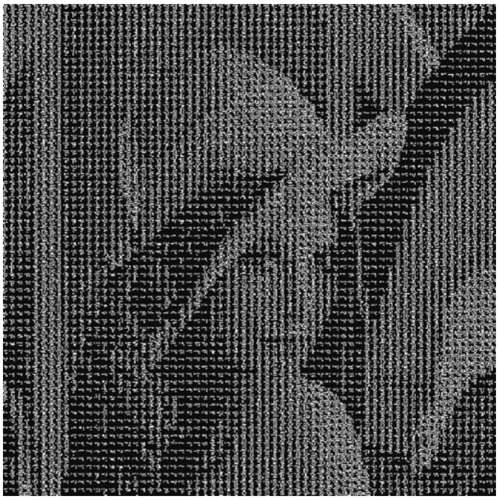
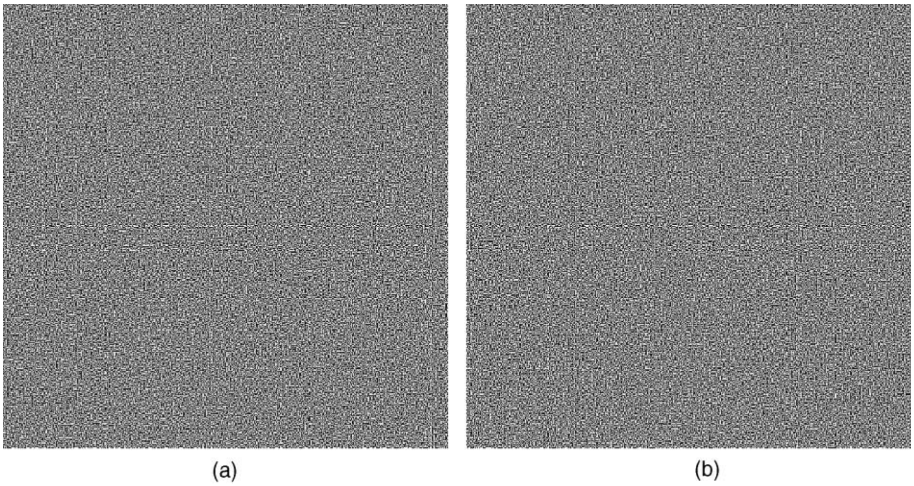


Fig. 14. The decoded image.

shown in Figs. 14 and 18, respectively. We observe that the two results, like that of the first experiment, are also acceptable. The size of either image in Figs. 14 and 18 is just 1/64 of that used by the Verheul and van Tilborg scheme. It is seen here that even when the number of gray levels in the original image reaches 256, the effect of our scheme is still satisfactory in the aspects of relative size increase and decoded image quality.

6. A person authentication application

Imagine a system for IC card authentication, which can be implemented by applying the proposed (2, 2)-threshold visual cryptography scheme for gray-level images. Instead of using existing credit cards, each user of the system has an IC card storing an image share that looks meaningless. This image share is generated by encrypting a portrait of the user. The process includes the cre-



(a) (b)

Fig. 13. The images of the shares. (a) The *share*1, (b) the *share*2.

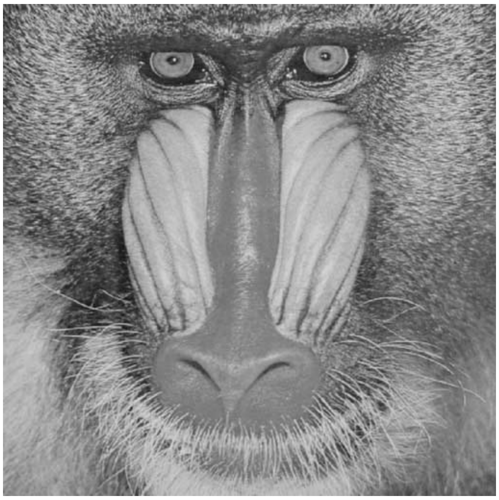


Fig. 15. The original image.

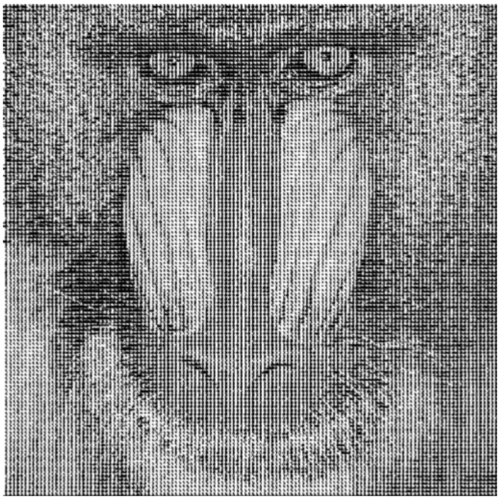


Fig. 16. The image after using SFCOD.

ation of two shares, one being a noisy *fixed* image which assumes the role of a public key in cryptography and is the same for all users; and the other share being a corresponding image which is the desired image share for each user stored in the IC card as a secret key. When the user goes shopping as a customer and pays a bill, the IC card is given to the clerk for decryption. Each shop owns the public key in a device and when the customer pays the bill, the clerk inserts the customer's IC card into the device and then the decoded image can be obtained by applying the "OR" operation to the two shares. Then the clerk can check the similarity between the decoded image revealed by the system and the appearance of the customer to authenticate the customer.

This person authentication application has several advantages. First, compared with authentication by passwords, it is more confidential. A 512 × 512 binary image has 2^{262144} decoding combinations, so the probability to obtain a correct guess of a person's secret key is very low. Second, it can, in addition to being used alone, be employed as a *visual* auxiliary tool for recognition of

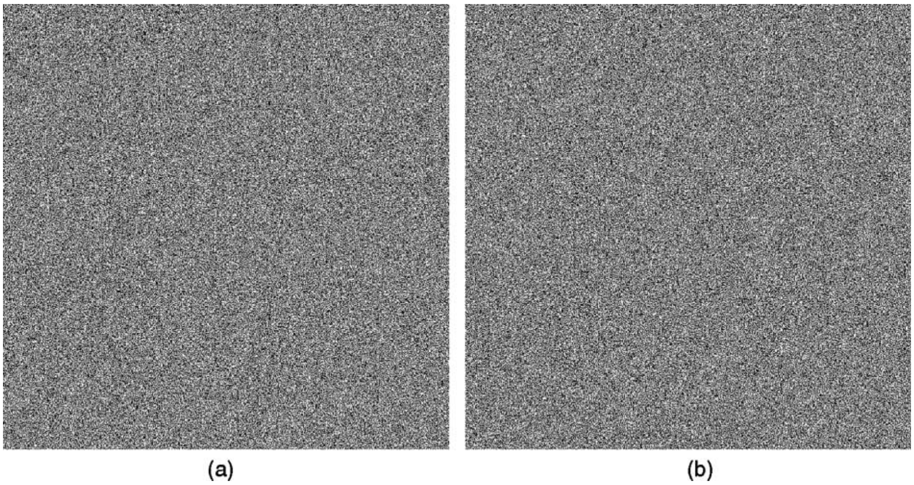


Fig. 17. The images of the shares. (a) The *share*1, (b) the *share*2.

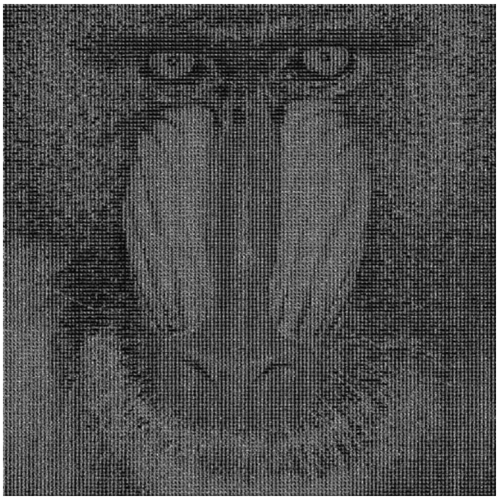


Fig. 18. The decoded image.

fingerprints and signatures because the proverb "to see is to believe" is still a habit of human beings. Many credit cards nowadays have the pictures of the owners of the cards for person identification. The function of this application here is the same but is with higher security. Finally, our system, which does not need intensive computation, may be implemented in an existing IC card system, and this makes the proposed scheme even more practical.

The above-mentioned person authentication application is just one of the many possible ones of visual cryptography schemes. With our proposed scheme, it is convenient to extend various applications of visual cryptography developed for binary images to those for gray-level images. This characteristic enlarges the applicability of our scheme.

7. Conclusions

Extension of visual cryptography for binary images to one for gray-level ones is useful for wider applications. In this study, we have developed a scheme that achieves this goal. An input gray-level image is first converted into an approximate binary image with the dithering technique, and a visual cryptography method for binary images is then applied to the resulting dither image. This scheme possesses the advantages of inheriting any developed cryptographic technique for binary images and having less increase of image size in ordinary situations. The decoded images can reveal most details of original images. Some experiments have been conducted to evaluate the effect of our proposed method. The experimental results prove that our approach is feasible. A possible application has also been proposed in this study. For further research, efficient visual cryptography for color images may be chosen as the next topic.

References

Ateniese, G., Blundo, C., De Santis, A., Stinson, D.R., 1996a. Visual cryptography for general access structures. Informat. Computat. 129, 86–106.

Ateniese, G., Blundo, C., De Santis, A., Stinson, D.R., 1996b. Constructions and bounds for visual cryptography. In: 23rd Internat. Colloquium on Automata, Languages and Programming Lecture Notes in Computer Science, vol. 1099, pp. 416–428.

Cox, I., Miller, M., Bloom, J., 2001. Digital Watermarking. Morgan Kaufmann Publishers, San Francisco.

De Bonis, A., De Santis, A., 2000. Randomness in visual cryptography. In: Proc. STACS 2000 LNCS 1770. Springer-Verlag, pp. 627–638.

Droste, S., 1996. New results on visual cryptography. In: Adv. Cryptol. CRYPTO'96 LNCS 1109. Springer-Verlag, Berlin, pp. 401–415.

Hofmeister, T., Krause, M., Simon, H.U., 1997. Contrast-optimal k out of n secret sharing schemes in visual cryptography. In: COCOON'97 LNCS 1276. Springer-Verlag, pp. 176–185.

Johnson, N.F., Duric, Z., Jajodia, S., 2000. Information Hiding: Steganography and Watermarking—Attackers and Countermeasures. Kluwer Academic Publishers, Dordrecht.

Naor, M., Shamir, A., 1995. Visual cryptography. In: Advances in Cryptology—EUROCRYPT'94 Lecture Notes in Computer Science, vol. 950, pp. 1–12.

Naor, M., Pinkas, B., 1997. Visual authentication and identification. In: Advances in Cryptology—CRYPTO'97 Lecture Notes in Computer Science, vol. 1294, pp. 322–336.

Shamir, A., 1979. How to share a secret. Communications of the Association for Computing Machinery 22 (11), 612–613.

Katzenbeisser, S., Petitcolas, F.A.P., 2000. Information Hiding Techniques for Steganography and Digital Watermarking. Artech House, MA.

Verheul, E.R., van Tilborg, H.C.A., 1997. Construction and Properties of k out of n visual secret sharing schemes. Designs Codes Cryptogr. 11, 179–196.

Zhang, Y., 1998. Space-filling curve ordered dither. Comput. Graphics 22 (4), 559–563.