JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 24, 1567-1577 (2008)

DOI:10.1688/JISE.2008.24.5.18

Short Paper

A Geometry-Based Secret Image Sharing Approach

CHIEN-CHANG CHEN AND WEN-YIN FU
Department of Computer Science
Hsuan Chuang University
Hsinchu, 300 Taiwan
E-mail: cchen34@hcu.edu.tw

A (k, n) secret image sharing method shares a protected image in n shared images and the protected image can be reconstructed by k shared images. This study solves the secret image sharing problem by a geometry secret sharing strategy named the Blakley scheme. The protected image is first partitioned into non-overlapping sets of k pixels. Each set of k pixels forms a point under a k-dimensional space and the set solution to each generated affine hyperplane, only intersecting at this point stores to the corresponding shared image. The reconstructed image, consistent with the protected image, is obtained from k shared images. Additionally, an efficient hyperplanes selection strategy is presented. Experimental results indicate that the proposed approach is efficient for secretly sharing a digital image.

Keywords: visual cryptography, secret sharing, (k, n) threshold, geometry, Blakley's secret sharing method

1. INTRODUCTION

Digital image can now be reproduced and spread easily. Therefore, preserving important images secretly is a major issue. Secret image sharing has been recently presented to solve this problem. Secret image sharing techniques generate several shared images from the protected image, and the protected image is reconstructed by enough different shared images.

Secret image sharing approaches can be split into two categories. One class of approaches piles up the shared images to obtain the reconstructed image. The other class of schemes mathematically calculates the reconstructed image. This study presents a secret image sharing approach based on mathematical calculation.

In first group, Naor and Shamir [7] first introduced the secret image sharing problem, and then presented a piling approach to share binary secret image. Blundo *et al.* [2] extended Naor and Shamir's technique to gray-level techniques. Lin and Tsai [6] adopted a dithering technique to convert a gray-level image into an approximate binary image, then adopted Naor and Shamir's binary method to share an image secretly. Hou [5] presented secret image sharing techniques for gray and color images. Viet and Kurosawa [10] presented an almost-perfect visual cryptography scheme with a reversing technique

Received November 7, 2006; revised March 29 & May 21, 2007; accepted June 27, 2007.
Communicated by Tzong-Chen Wu.

that allowing each participant to reverse its transparencies. Cimato *et al.* [4] further improved on Viet and Kurosawa's method to present a perfect visual cryptography scheme with reversing technique.

In other group, Thien and Lin [9] employed a mapping key to permute the secret image, and then applied Shamir's [8] secret sharing method to generate shared images. Wu *et al*. [12] further extended Thien and Lin's [9] method to embed shared images into other ordinary images. Chen and Lin [3] developed a progressive secret sharing approach to improve the quality of the reconstructed image by increasing the number of shared images. Wang and Su [11] further adopted the Galois field and Huffman code to improve Thien and Lin's method in terms of storage and efficiency.

This study presents a new mathematically secret image sharing method. The proposed method generates shared images by the Blakley's secret sharing method [1]. Blakley's (k, n) secret sharing method is a geometry-based approach that depicts a point in a k-dimensional space, and each share represents one hyperplane intersecting at this point. The proposed secret image sharing method has the following properties:

1. No additional information is required except thresholds (k, n).
2. The protected image can be perfectly reconstructed by gathering any k different shared images.
3. The protected image cannot be reconstructed when gathering fewer than k shared images.
4. The generated shared images have the same size as the protected image.

The rest of this paper is organized as follows. Section 2 reviews Blakley's secret sharing scheme. Section 3 describes the proposed geometry-based secret image sharing approach that includes a sharing algorithm and a recovering algorithm. A fast and secure method to generate shared images is also presented. Section 4 presents some experimental results of the proposed approach and analyzes its security. Conclusions are finally drawn in section 5.

2. REVIEW OF BLAKLEY'S SECRET SHARING SCHEME

A secret sharing scheme shares a message among n trustees, of whom any k can recover the secret message. Blakley's secret sharing method is adopted to share protected images in the proposed method, and is therefore briefly introduced in this section.

Blakley [1] adopted geometry to solve the secret sharing problem. The secret message is a point in a k-dimensional space, and n shares are affine hyperplanes that intersect at this point. The set solution $x = (x_1, x_2, \ldots, x_k)$ to an equation $a_1x_1 + a_2x_2 + \ldots + a_kx_k = b$ forms an affine hyperplane. The secret, the intersection point, is calculated by finding the intersection of any k of these planes.

Fig. 1 shows an example of (k, n) Blakley's scheme with $k = 2$ as a two-dimensional plane and $n = 3$ as three shares. The secret is a 2-dimensional point T, and three shares (L_1, L_2, L_3) denotes lines with different parameters (a_1, a_2, b) passing through the point T. When gathering any two lines, for instance L_1 and L_2, the secret T can be acquired by finding the point where they intersect.

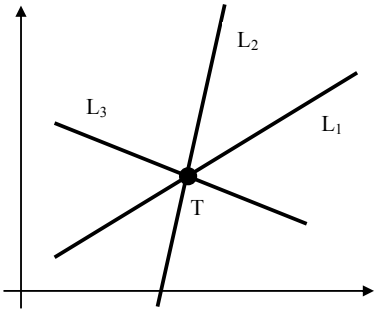


Fig. 1. A two-dimensional Blakley's secret sharing scheme.

3. THE PROPOSED SECRET IMAGE SHARING APPROACH

A secret image sharing approach includes two algorithms, a sharing algorithm to generate shared images from a protected image, and a recovering algorithm to calculate the reconstructed image from shared images. Section 3.1 introduces the proposed sharing algorithm, and section 3.2 introduces the proposed recovering algorithm. Section 3.3 presents a hyperplane selection strategy to acquire secure hyperplanes efficiently.

3.1 The Sharing Algorithm

This section introduces the secret sharing of protected images by Blakley's secret sharing scheme. When applying the Blakley's multi-dimensional scheme to the visual cryptography problem, two parameters k and n representing the required shared images to recover the secret image and the generated shared images, should first be determined. An image is then partitioned into non-overlapping sets of k pixels, each of which forms a point under a k-dimensional space. One affine hyperplane passing this point constitutes part of each generated shared image. Assume that n shared images $S_1, S_2, \ldots, S_n$ are to be generated from the protected image S, and that gathering k or more shared images can recover the protected image S as a (k, n) threshold. The sharing algorithm is depicted in Fig. 2 and is described below.

Step 1: Select thresholds (k, n).
Step 2: Partition the protected image into non-overlapping sets of k pixels.
Step 3: For each set of k pixels,
- **3.1** These k pixels form a k-dimensional point $x = (x_1, x_2, \ldots, x_k)$.
- **3.2** Randomly select n different solution sets $(a_1, a_2, \ldots, a_k, b)$ such that equation $a_1x_1 + a_2x_2 + \ldots + a_kx_k = b$ is satisfied.
- **3.3** Adjust each set of parameters $(a_1, a_2, \ldots, a_k, b)$ to pre-defined bits.

Step 4: Store each set of bits in its shared image.

In the proposed sharing algorithm, the randomly selected equations in step 3.2 must satisfy one requirement that these equations only intersect at one point. Each set solution in step 3.2 is then adjusted to pre-defined bits in step 3.3, and stored into one shared image as shown in step 4. The assignment of pre-defined bits determines the size of each shared image.

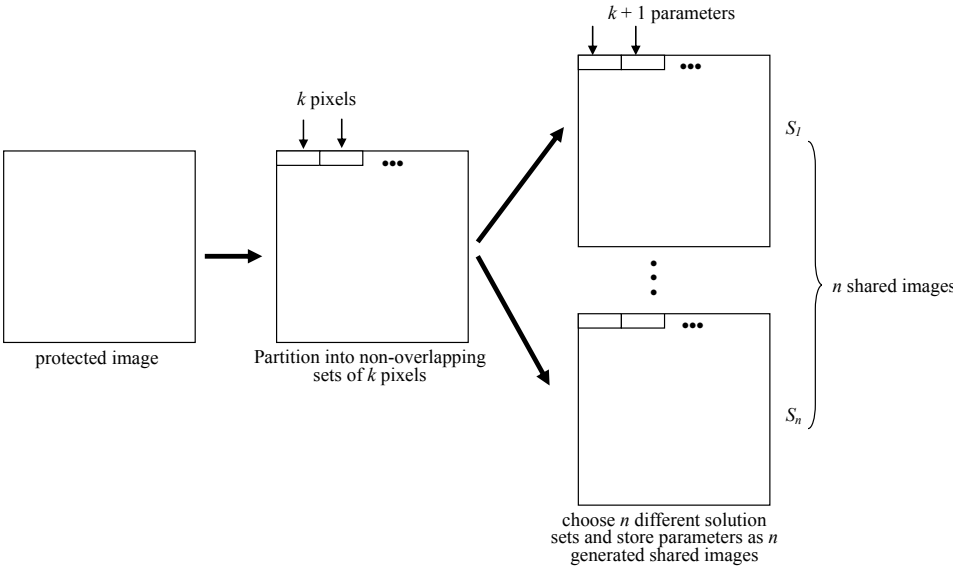


Fig. 2. The proposed sharing algorithm.

The following assignment is adopted to adjust all shared images with the same size as the protected image. In a (k, n) threshold, each set contains k pixels (bytes). The k parameters (a_1, a_2, …, a_k) and one constant parameter b are stored by these k bytes to keep the image size invariant. Thus, k bytes are employed to store these $k + 1$ parameters. The following assignment is adopted to satisfy this requirement. Parameter b is stored by experiments as $6 + 2 \times k$ bits, and each parameter in (a_1, a_2, …, a_{k-1}) is stored as $\lfloor (8 \times k - (6 + 2 \times k))/k \rfloor$ bits, and parameter a_k is stored as $8 \times k - \lfloor (8 \times k - (6 + 2 \times k))/k \rfloor \times (k - 1) - (6 + 2 \times k)$ bits. The parameters in step 3.3 are adjusted to satisfy this rule. Representing parameter b by $6 + 2 \times k$ bits fits the condition of pixels near the boundary in an n-dimensional space.

For example, thresholds (k, n) = (2, 3) and (3, 5) were selected in experiments in this study. In (k, n) = (2, 3), the image pixels are partitioned into sets of 2 pixels. Our previous description defines $6 + 2 \times k = 10$, $\lfloor (8 \times k - (6 + 2 \times k))/k \rfloor = 3$ and $8 \times k - \lfloor (8 \times k - (6 + 2 \times k))/k \rfloor \times (k - 1) - (6 + 2 \times k) = 3$. Three bits are then adopted to represent parameters a_1 or a_2, and 10 bits are adopted to represent parameter b. A parameter represented by p bits has a range of -2^{p-1} to $2^{p-1} - 1$. However, to maintain security, parameters a_1 and a_2 should not be 0, because they could be easily guessed. Therefore, parameters a_1 and a_2 are in the range between − 4 and 4, excluding 0. Since parameters a_1 and a_2 can be positive or negative numbers, parameter b has the range 0 and 1023 to eliminate duplication of the range between − 512 and 511.

In the threshold (k, n) = (3, 5), the image pixels are partitioned into sets of 3 pixels. Our definition calculates three numbers $6 + 2 \times k = 12$, $\lfloor (8 \times k - (6 + 2 \times k))/k \rfloor = 4$ and $8 \times k - \lfloor (8 \times k - (6 + 2 \times k))/k \rfloor \times (k - 1) - (6 + 2 \times k) = 4$. Consequently, 4 bits are adopted to represent parameters (a_1, a_2, a_3) and 12 bits are adopted to represent parameter b. Thus, parameters a_1, a_2 and a_3 are in the range between − 8 and 8, excluding 0. Parameter b is the range between 0 and 4095.

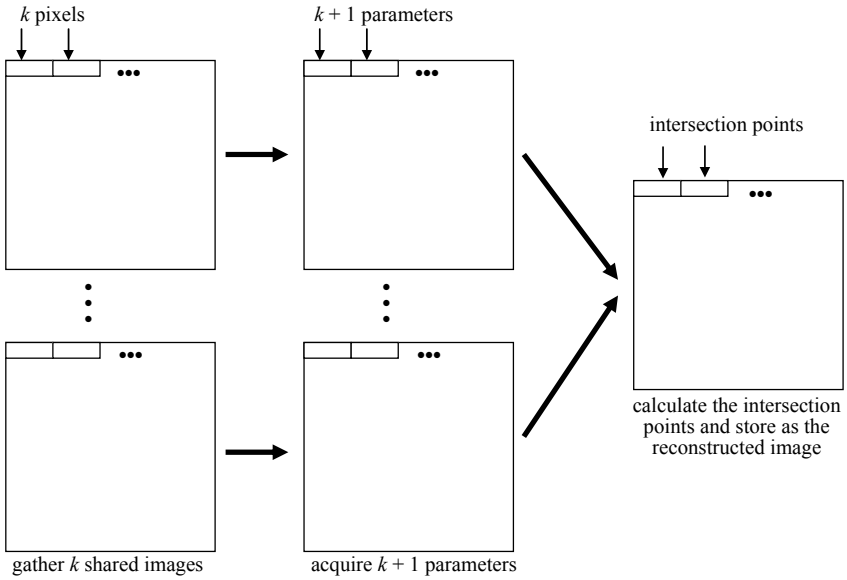


Fig. 3. The proposed recovering algorithm.

3.2 The Recovering Algorithm

The recovering algorithm calculates the reconstructed image from k or more shared images. The same thresholds (k, n) should be adopted for recovering the reconstructed image as in the sharing algorithm. Fig. 3 shows the recovering algorithm, which is described as follows.

Step 1: Adopt the same thresholds (k, n) and k shared images.
Step 2: Partition each shared image into non-overlapping sets of k pixels.
Step 3: Retrieve $k + 1$ parameters (a_1, a_2, …, a_k, b) from k pixels by the method discussed in section 3.1.
Step 4: Identify the intersection point of these k hyperplanes constructed by the parameters calculated in step 3.
Step 5: Store the coordinate of the intersection point under a k-dimensional space as k pixels in the reconstructed image.
Step 6: Perform steps 2-5 on all sets to acquire the reconstructed image.

3.3 A Fast and Secure Hyperplanes Selection Strategy

This section describes an efficient parameters selection method for acquiring secure hyperplanes. Section 3.1 decomposes the secret image into sets of k pixels, where each acting as a secret point in a k-dimensional space. Each set of n shared hyperplanes intersects at only one k-dimensional point. In contrast, each generated hyperplane passes many points, but only one is the secret point. Thus, a hyperplane passing more points in a k-dimensional space is more secure for the adversary to guess the secret point. The

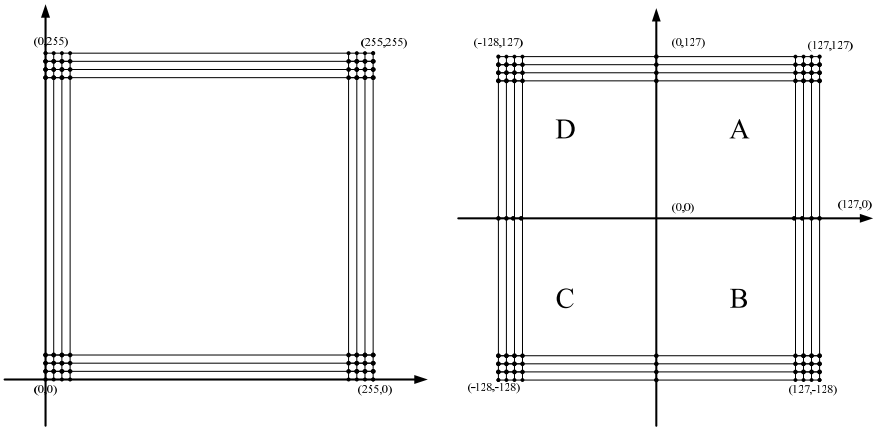


Fig. 4. (a) Four regions defined in a two-dimensional space; (b) Shifted space.

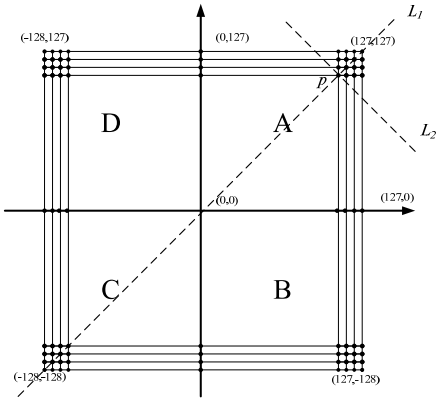


Fig. 5. Example of two extreme lines L_1 and L_2 passing point p.

proposed method shifts the coordinates of each point to the center. In a two-dimensional space, each point is shifted by the center (128, 128). Moreover, a three-dimensional space shifts all points by (128, 128, 128). Fig. 4 (a) shows the original two-dimensional space and Fig. 4 (b) shows the shifted space, in which (128, 128) is subtracted from each point. The shifted space is then partitioned to four areas. Hyperplane selection is important for points close to the boundary of the space. Fig. 5 shows two extreme selected lines L_1 and L_2 passing the secret point p on a two-dimensional space. Line L_1 is better than line L_2, since line L_1 passes many points than line L_2 does. Namely, selecting line L_1 is more secure than selecting line L_2. Thus, an efficient method to select secure lines is presented.

The following steps generate one secure line on a shifted two-dimensional space efficiently.

1. Locate the secret point in shifted space.
2. Randomly choose one point far away from the secret point.

3. Adjust the slope of a line passing these two points to the parameters nearest to them.
4. Build the line from the slope selected in step 3 and the secret point.

Fig. 6 shows the selection method in step 2, where the gray area denotes the selected candidates for the point p. For a point in area A, the selected point locates at the boundary of three other areas. The following example illustrates this selection. Assume the secret point is (252, 252), in which case its shifted coordinates are (124, 124), which are the shifted coordinates of the center (128, 128). The point p is located in area A. Therefore, a point in the boundary of areas B, C, and D is randomly selected. If the randomly selected point is located at (– 124, 30), then the slope of the line passes these two points as (124 − 30)/(124 − (– 124)) = 94/248. Since parameters a_1 and a_2 are assigned in the range between − 4 and + 4, the nearest slope is 1/3. Consequently, the selected line is acquired as $-x + 3y = 248$. In a (2, n) threshold, n lines are randomly selected from the above selection steps. In particular, the selection slope would prevent the adoption 0 and ∞. Additionally, each pair of selected lines should only intersect at the secret point p.

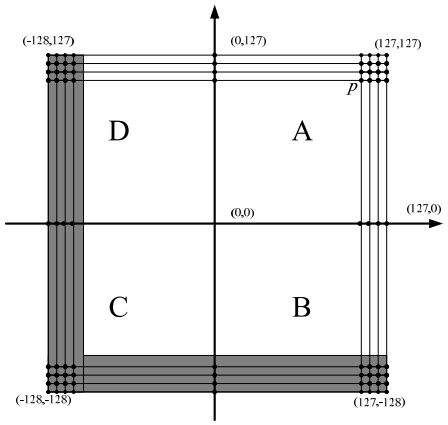


Fig. 6. Selected candidates for the secret point p.

An image may have some areas, in which the gray values are almost the same. Thus, the hyperplane parameters must prevent the adversary to guess the area by two neighboring blocks parameters with different parameters. When the difference of shifted coordinates between two neighboring blocks are lower than a pre-determined threshold, the hyperplane parameters are chosen randomly from the slope used in previous block or new generated parameters. Consequently, two parallel hyperplanes are existed in neighboring blocks randomly to let the adversary hard to figure out the area.

Notably, the parameters in a three-dimensional or higher dimensional space can also be acquired by similar steps. The only difference is the number of selected points. In a (k, n) threshold, n hyperplanes are created in a k-dimensional space, and k hyperplanes reconstruct the secret point. However, building a hyperplane in a k-dimensional space needs k points, thus, one hyperplane can be built by randomly selecting (k − 1) points and the secret point. Each generated k hyperplanes must intersect only at the secret point.

4. EXPERIMENTAL RESULTS AND SECURITY COMPARISONS

4.1 Experimental Results

This section presents the experimental results of the proposed geometry-based secret image sharing method. Two thresholds, (2, 3) and (3, 5) are experimented. Threshold (2, 3) indicates that 3 shared images are randomly generated, and the reconstructed image can be calculated by gathering two images. The protected images are JET and LENNA with size 256 × 256. Fig. 7 shows the experimental results of threshold (2, 3). Fig. 7 (a) shows the protected image JET, and Fig. 7 (e) depicts the reconstructed image. Both of these two images are consistent. Figs. 7 (b)-(d) show three randomly generated shared images.

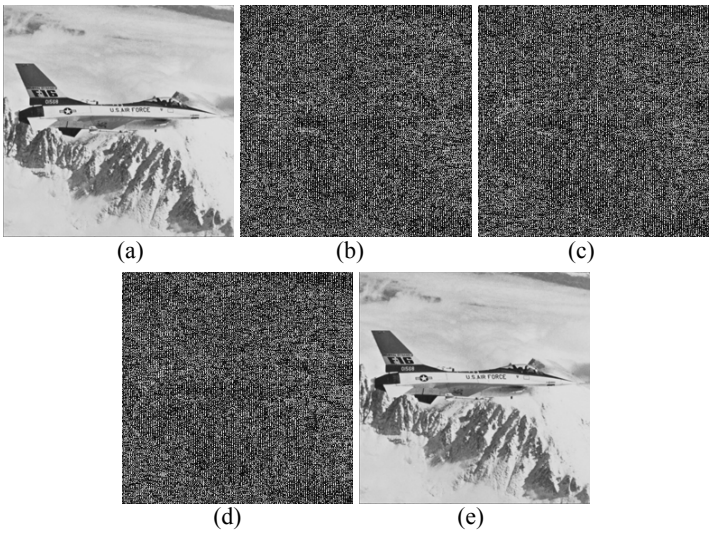
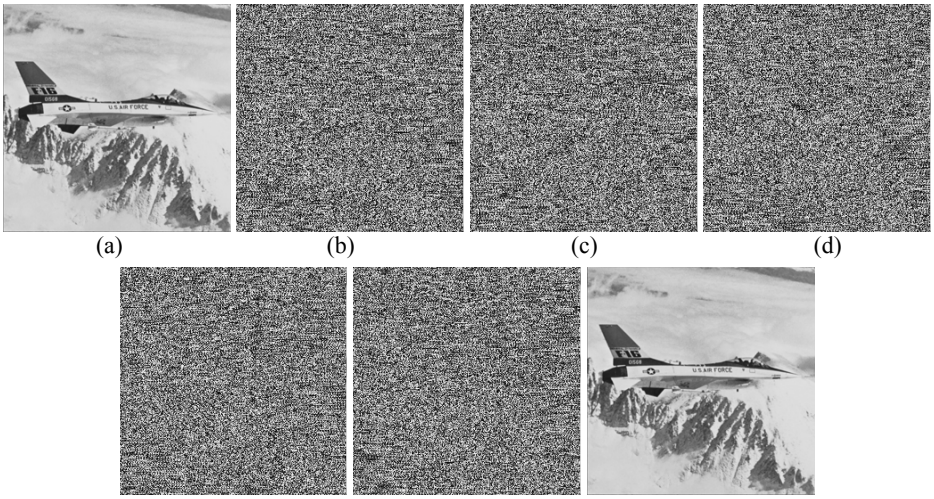


Fig. 7. (a) The protected image; (b)-(d) Three shared images; (e) The reconstructed image.

Fig. 8 shows the experimental results of threshold (3, 5). Fig. 8 (a) shows the protected image, Figs. 8 (b)-(f) show five randomly generated shared images, and Fig. 8 (g) displays the reconstructed image, which is also consistent with the protected image.

Fig. 9 shows the experimental results of selecting threshold (3, 5) on another image, LENNA. Fig. 9 (a) shows the protected image, and Fig. 9 (g) denotes the reconstructed image. Both of these two images are consistent. Figs. 9 (b)-(f) show five randomly generated shared images.

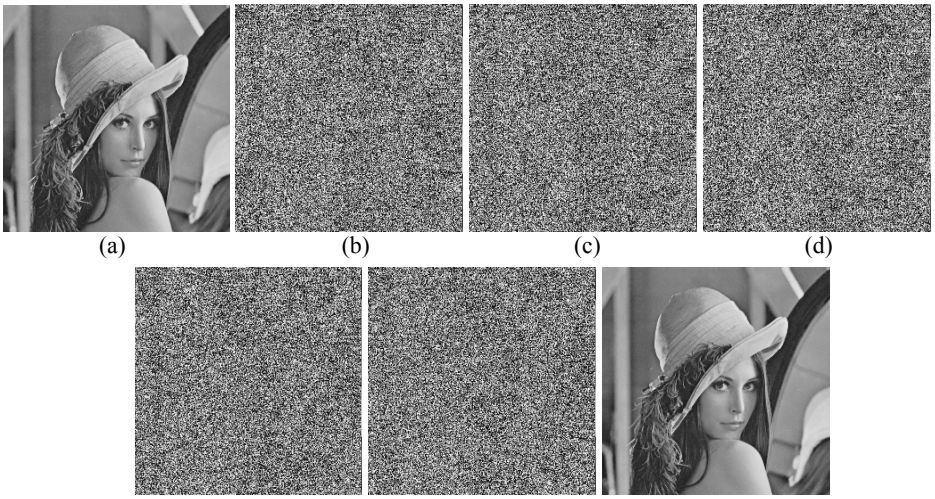
These experimental results indicate two properties. First, these generated shared images constitute random noise because of their randomly selected hyperplane parameters. Thus, an adversary cannot figure out content of the protected image from only one shared image. Second, the proposed approach generates randomly shared images directly. The proposed approach is thus better than Thien and Lin's approach, which requires a permutation key to break the protected image into a noise-like permutated sequence.



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

(e) (f) (g)

Fig. 8. (a) The protected image; (b)-(f) Five shared images; (g) The reconstructed image.



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

(e) (f) (g)

Fig. 9. (a) The protected image; (b)-(f) Five shared images; (g) The reconstructed image.

4.2 Security Discussion

Finally, the security of the proposed approach is analyzed. The proposed sharing algorithm shares the secret image to n shared images, where the reconstructed image can be calculated by collecting k shared images. An adversary containing one shared image has parameters for many k dimensional hyperplanes, and each hyperplane contains one secret point. Thus, the guessing probability for the adversary is the rate of guessing the correct point.

Table 1. Guessing probability of an image with size 256 × 256 under different space.

Space	Hyperplane passing points	Guessing probability
Two-dimensional	≥ 64	$\leq \left(\frac{1}{64}\right)^{\frac{256\times256}{2}}$
Three-dimensional	≥ 6400	$\leq \left(\frac{1}{6400}\right)^{\frac{256\times256}{3}}$

A line in a two-dimensional space passes at least 64 points by parameters restriction in section 3.1. For an image with size 256 × 256, the probability of guessing the secret image is below $\left(\frac{1}{64}\right)^{\frac{256\times256}{2}}$. One hyperplane passes at least 6,400 points when $k = 3$, *i.e.* in three-dimensional space. Thus the probability for guessing the secret image is below $\left(\frac{1}{6400}\right)^{\frac{256\times256}{3}}$. Table 1 compares two and three dimensional spaces.

When threshold $k > 2$, the attacker guessing probability performs better than Lin and Thien's approach of $\left(\frac{1}{251}\right)^{\frac{256\times256}{k}}$. Moreover, Lin and Thien's approach has the disadvantages of imperfect reconstruction or irregular sizes of shared images. However, the proposed method can reconstruct the original image perfectly, and ensures that each share has the same size as the original image. Moreover, unlike previous method, the proposed method does not require random permutation.

5. CONCLUSION

This study presents a geometry-based secret image sharing method. The proposed method randomly generates all n shared images, and prevents recovery of the reconstructed image from fewer than k shared images. Additionally, an efficient strategy is presented to generate shared images fast and securely. The proposed method, unlike previous methods, does not need an arrangement key, also exhibit better characteristic. Experimental results and brief comparisons indicate the excellent properties of the proposed method.

ACKNOWLEDGEMENTS

The authors would like to thank the reviewers for valuable comments.

REFERENCES

1. G. R. Blakley, "Safeguarding cryptographic keys," in *Proceedings of the AFIPS National Computer Conference*, Vol. 48, 1979, pp. 313-317.

2. C. Blundo, A. D. Santis, and M. Naor, "Visual cryptography for grey level images," *Information Processing Letters*, Vol. 75, 2000, pp. 255-259.
3. S. K. Chen and J. C. Lin, "Fault-tolerance and progressive transmission of images," *Pattern Recognition*, Vol. 38, 2005, pp. 2466-2471.
4. S. Cimato, A. D. Santis, A. L. Ferrara, and B. Masucci, "Ideal contrast visual cryptography schemes with reversing," *Information Processing Letters*, Vol. 93, 2005, pp. 199-206.
5. Y. C. Hou, "Visual cryptography for color images," *Pattern Recognition*, Vol. 36, 2003, pp. 1619-1629.
6. C. C. Lin and W. H. Tsai, "Visual cryptography for gray-level images by dithering techniques," *Pattern Recognition Letters*, Vol. 24, 2003, pp. 349-358.
7. M. Naor and A. Shamir, "Visual cryptography," in *Proceedings of Advances in Cryptology – EURPOCRYPT*, LNCS 950, 1995, pp. 1-12.
8. A. Shamir, "How to share a secret," *Communications of the ACM*, Vol. 22, 1979, pp. 612-613.
9. C. C. Thien and J. C. Lin, "Secret image sharing," *Computers and Graphics*, Vol. 26, 2002, pp. 765-770.
10. D. Q. Viet and K. Kurosawa, "Almost ideal contrast visual cryptography with reversing," in *Proceeding of Topics in Cryptology – CT-RSA*, LNCS 2964, Springer, 2004, pp. 353-365.
11. R. Z. Wang and C. H. Su, "Secret image sharing with smaller shadow images," *Pattern Recognition Letters*, Vol. 27, 2006, pp. 551-555.
12. Y. S. Wu, C. C. Thien, and J. C. Lin, "Sharing and hiding secret images with size constraint," *Pattern Recognition*, Vol. 37, 2004, pp. 1377-1385.

Chien-Chang Chen (陳建彰) received the B.S. degree from Department of Computer and Information Science at Tung Hai University, Taichung, in 1991, and the Ph.D. degree in Computer Science from National Tsing Hua University in 1999. He is currently an Assistant Professor at the Department of Computer Science, Hsuan Chuang University. His research interests include image authentication, watermarking, vision cryptography, and texture analysis.

Wen-Yin Fu (傅文殷) was born in Taichung, Taiwan, R.O.C. in 1981. He received the M.B.A. degree in Information Management from Hsuan Chuang University, Taiwan, R.O.C. in 2005. He is currently an Information Engineer in Aerospace Industrial Development Corporation (AIDC), Taiwan, R.O.C. His major research interests include secret sharing and visual cryptography.