A Reliable (k, n) Image Secret Sharing Scheme

Li Bai
ECE Department
Temple University
Philadelphia, PA, U.S.A.
lbai@temple.edu

Abstract

This paper presents a reliable image secret sharing method which incorporates two k-out-of-n secret sharing schemes: i) Shamir's secret sharing scheme and ii) matrix projection secret sharing scheme. The technique allows a colored secret image to be divided as n image shares so that: i) any k image shares ($k \leq n$) are sufficient to reconstruct the secret image in the lossless manner and ii) any $(k-1)$ or fewer image shares cannot get enough information to reveal the secret image. It is an effective, reliable and secure method to prevent the secret image from being lost, stolen or corrupted. In comparison with other image secret sharing methods, this approach's advantages are its large compression rate on the size of the image shares, its strong protection of the secret image and its ability for the realtime processing.

Keywords: image processing, secret sharing, matrix projection.

1 Introduction

The effective and secure protections of sensitive information [18] are primary concerns in commercial, medical and military systems (*e.g.* communication systems or network storage systems). Needless to say, it is also important for any information process to ensure data is not being tampered. Encryption methods are one of the popular approaches to ensure the integrity and secrecy of the protected information. However, one of the critical vulnerabilities of encryption techniques is the single-point-failure. For example, the secret information cannot be recovered if the decryption key is lost or the encrypted content is corrupted during the transmission. To address these reliability problems, in particular for large information content items such as secret images (say satellite photos or medical images), an image *secret sharing scheme* (SSS) is a good alternative to remedy these types of vulnerabilities.

Blakley [4, 11] and Shamir [16] invented two (k, n) threshold-based SSS independently in 1979. The general idea behind "secret sharing" is to distribute a secret (*e.g.*, encryption/decryption key) to n different participants so that any k participants can reconstruct the secret, and any $(k-1)$ or fewer participants cannot reveal anything about the secret. Karnin *et al.* [9] suggested the concept of *perfect secret sharing* (PSS) where zero information of the secret is revealed for an unqualified group of $(k-1)$ or fewer members. Apparently, there is a subtle difference between the unqualified group cannot obtain any information about the secret and the unqualified group cannot reconstruct the secret with some information. For example, an unqualified group may know information about the secret as an even number, but the group still cannot discover the exact value of the secret. Specifically, Karnin *et al.* used a term referred as information entropy (a measurement of the uncertainty of the secret), denoted as $H(s)$ where s is a secret shared among n participants. The claim of PSS schemes must satisfy the following:

1. a qualified coalition of k or more participants, C can reconstruct the secret(s), s:
   $H(s|C) = 0 \qquad \forall |C| \geq k,$
2. an unqualified coalition of $(k-1)$ or few participants, C has no information about the secret(s), s:
   $H(s|C) = H(s) \qquad \forall |C| < k.$

For these requirements in PSS schemes, a secret has zero uncertainty if the secret can be discovered by k or more participants. On the contrary, the secret, in PSS schemes, remain the same uncertainty for $(k-1)$ or fewer members. Therefore, there is no information exposed to the $(k-1)$ or fewer members.

When exposed information is proportional to the size of the unqualified coalition, these types of SSS are referred as a *ramp secret sharing* (RSS) [10, 14, 15, 17]. Various research papers are devoted on the topics of PSS schemes [1, 3] and RSS schemes [5, 7, 8].

Proceedings of the 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC'06)
0-7695-2539-3/06 $20.00 © 2006 IEEE

IEEE COMPUTER SOCIETY

Naor and Shamir [12, 13] extended the secret sharing concept into image research, and referred it as visual cryptography. Visual cryptography is a PSS scheme, and requires stacking any k image shares (or shadow images) to show the original image without any cryptographic computation. They are not applicable for lossless image recovery due to: i) image shares have larger image size compared to the size of the original secret image and ii) the contrast ratio in the reconstructed image is quite poor. A better image secret sharing approach was presented by Thien and Lin [18]. With some cryptographic computation, they cleverly used Shamir's SSS to share a secret image. The method significantly reduces the size of the image shares to become $1/k$ of the size of the secret image, and the secret image can be reconstructed with good quality. A drawback, in terms of security, requires that the image is permutated by a key before the image share can be computed.

We propose to enhance Thien and Lin's image SSS by incorporating another secret sharing method – matrix projection SSS. The size of image shares is increased, but is still significantly less than the size of the secret image. However, this approach provides a better security measure to protect the image content and an adaptive method to reconstruct image in lossless manner. This method also allows realtime processing in colored images.

The rest of paper is organized as follows: a brief review is given in section 2 about secret sharing schemes including matrix projection SSS and Thien and Lin's image SSS. Our technique is proposed in section 3 with its advantages and the conclusion given in section 4.

2 Review of Secret Sharing Schemes

We describe several (k, n) threshold-based SSSs and describe how a secret and an image is shared among n participants. These schemes are briefly described in this section with their interesting features.

2.1 Shamir's Secret Sharing Scheme

Shamir [16] developed the idea of a (k, n) threshold-based secret sharing technique ($k \leq n$). The technique allows a polynomial function of order $(k-1)$ constructed as,

$$f(x) = d_0 + d_1 x + d_2 x^2 + \ldots + d_{k-1} x^{k-1} \pmod{p},$$

where the value d_0 is the secret and p is a prime number. The secret shares are the pairs of values (x_i, y_i) where $y_i = f(x_i)$, $1 \leq i \leq n$ and $0 < x_1 < x_2 \ldots < x_n \leq p-1$.

The polynomial function $f(x)$ is destroyed after each shareholder possesses a pair of values (x_i, y_i) so that no single shareholder knows the secret value d_0. In fact, no groups of $k-1$ or fewer secret shares can discover the secret d_0. On the other hand, when k or more secret shares are available, then we may set at least k linear equations $y_i = f(x_i)$ for the unknown d_i's. The unique solution to these equations shows that the secret value d_0 can be easily obtained by using Lagrange interpolation [16].

Shamir's SSS is regarded as a PSS scheme because knowing even $(k-1)$ linear equations doesn't expose any information about the secret.

2.2 Thien and Lin's Image Secret Sharing Scheme

Thien and Lin [18] proposed a (k, n) threshold-based image SSS by cleverly using Shamir's SSS [16] to generate image shares. The essential idea is to use a polynomial function of order $(k-1)$ to construct n image shares from an $l \times l$ pixels secret image (denoted as I) as,

$$\begin{aligned} S_x(i,j) &= I(ik+1, j) + I(ik+2, j)x \ldots + \\ & \quad I(ik+k, j)x^{k-1} \pmod{p} \end{aligned} \tag{1}$$

where $0 \leq i \leq \lfloor \frac{l}{k} \rfloor$ and $1 \leq j \leq l$. This method reduces the size of image shares to become $1/k$ of the size of the secret image. Any k image shares are able to reconstruct every pixel value in the secret image. Thien and Lin also provided some research insights for lossless image recovery using their technique. They further introduced the possibility of a steganography approach [18, 19] by hiding image shares into host images.

An example of (2, 4) image secret share construction process is illustrated in Figure 1 where $k = 2$ and $n = 4$. According to the technique, a first order polynomial function can be created as

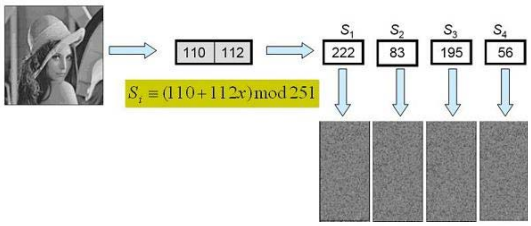

Figure 1. Secret Sharing Process for Lena Image

$$S_x(i,j) = (110 + 112x) \pmod{251} \tag{2}$$

where 110 and 112 are the first two pixel values in the Lena image. For our four participants, we can randomly pick four x values, and substitute them into the polynomial function

Proceedings of the 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC'06)
0-7695-2539-3/06 $20.00 © 2006 IEEE
IEEE COMPUTER SOCIETY

by setting p value to be 251 which is the largest prime number less than 255 (maximum gray image value). Four shares are computed as (1, 222), (2, 83), (3, 195) and (4, 56). They become the first pixel in four image shares. The second pixel is computed in the same manner by constructing another first order polynomial function using next two pixels in the Lena image. This process continues until all pixels are encoded. Four image shares are the bottom right images shown in Figure 1, and the size of each image share is half $(1/2)$ size of the original image. None of the image shares appear to reveal information about the secret image. However, the pixel values in a natural image are not random because the neighboring pixels often have equal or close values. It is evident that the first two pixel values (110 and 112) are very close to each other. That creates the possibility that one image secret share may be used to recover the secret image by assuming the neighboring pixels have the same values in the first order polynomial function.

Since Thien and Lin's method reduces the size of image shares to become $1/k$ of the size of the secret image, the scheme cannot be qualified as a "perfect" image SSS [6, 9]. In fact, this method is a multiple-secret "ramp" SSS [7]. In other words, the information about the secret exposed is proportional to the number of shares available until the number of shares becomes k or more. In addition, the pixel values in a natural image are not random because the neighboring pixels often have equal or close values. A secret image can be possibly recovered from less than k image shares because neighboring pixels are highly correlated. To address these security issues, Thien and Lin suggested an idea by permutating the order of pixels (with a permutation key) in the secret image before the image shares are computed. Conversely, the secret image can still be reconstructed from any k image shares by solving the permutated image and applying inverse-permutation using the permutation key. Nevertheless, the permutation key becomes the single-point-failure in the system because the key can get lost or corrupted. This scheme also prevents realtime processing because the permuted image has to be obtained before the secret image can be reconstructed.

2.3 Matrix Projection Secret Sharing Scheme

Bai [2] developed a SSS using matrix projection. The idea is based upon the invariance property of matrix projection. This scheme can be used to share multiple secrets, and detail of the scheme can be found in [2]. Here, we briefly describe the procedure in two phases:

- Construction of Secret Shares from secret matrix S
  1. Construct a random $m \times k$ matrix A of rank k where $m > 2(k-1)-1$,
  2. Choose n linearly independent $k \times 1$ random vectors x_i,
  3. Calculate share $v_i = (A \times x_i) \pmod{p}$ for $1 \leq i \leq n$.
  4. Compute $\mathbb{S} = (A(A'A)^{-1}A') \pmod{p}$,
  5. Solve $R = (S - \mathbb{S}) \pmod{p}$,
  6. Destroy matrix A, x_is, $\mathbb{S}$, S, and
  7. Distribute n shares v_i to n participants and make matrix R publicly known.
- Secret Reconstruction
  1. Collect k shares from any k participants, say the shares are $v_1, v_2, \ldots, v_k$ and construct a matrix $B = \begin{bmatrix} v_1 & v_2 & \ldots & v_k \end{bmatrix}$.
  2. Calculate the projection matrix $\mathbb{S} = (B(B'B)^{-1}B') \pmod{p}$,
  3. Verify that $\mathrm{tr}(\mathbb{S}) = k$, and
  4. Compute the secret $S = (\mathbb{S} + R \pmod{p}$.

A simple $(2, 4)$ threshold-based example is shown for $p = 251$ and the secret matrix

$$S = \begin{bmatrix} 2 & 3 & 1 & 2 \\ 5 & 4 & 6 & 1 \\ 8 & 9 & 7 & 2 \\ 3 & 4 & 1 & 2 \end{bmatrix}.$$

To construct the shares, we choose a 4×2 random matrix A of rank 2 that

$$A = \begin{bmatrix} 10 & 1 \\ 7 & 2 \\ 8 & 4 \\ 1 & 1 \end{bmatrix}.$$

The values of $m = 4$ and $k = 2$ satisfy the condition of secret sharing where $m > 2(k-1)-1$. Choose $n = 4$ linearly independent vectors as

$$x_1 = \begin{bmatrix} 1 \\ 17 \end{bmatrix}, \quad x_2 = \begin{bmatrix} 1 \\ 7 \end{bmatrix}, \quad x_3 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \text{and} \quad x_4 = \begin{bmatrix} 1 \\ 9 \end{bmatrix}.$$

Next we compute $v_i = Ax_i$ for $i = 1, 2, 3, 4$,

$$v_1 = \begin{bmatrix} 27 \\ 41 \\ 76 \\ 18 \end{bmatrix}, \quad v_2 = \begin{bmatrix} 17 \\ 21 \\ 36 \\ 8 \end{bmatrix}, v_3 = \begin{bmatrix} 11 \\ 9 \\ 12 \\ 2 \end{bmatrix}, \quad \text{and} \quad v_4 = \begin{bmatrix} 19 \\ 25 \\ 44 \\ 10 \end{bmatrix}.$$

The projection matrix $\mathbb{S}$ is

$$\mathbb{S} = (A(A'A)^{-1}A') \pmod{19} = \begin{bmatrix} 53 & 87 & 88 & 175 \\ 87 & 137 & 199 & 100 \\ 88 & 199 & 119 & 46 \\ 175 & 100 & 46 & 195 \end{bmatrix},$$

Proceedings of the 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC'06)
0-7695-2539-3/06 $20.00 © 2006 IEEE
IEEE COMPUTER SOCIETY

then the remainder matrix R is equal to

$$R = (S - \mathbb{S}) \pmod{19} = \begin{bmatrix} 51 & 84 & 87 & 173 \\ 82 & 133 & 193 & 99 \\ 80 & 190 & 112 & 44 \\ 172 & 96 & 45 & 193 \end{bmatrix}.$$

The matrix R is made public. We can destroy A, x_is, $\mathbb{S}$ and S, then we distribute three v_i shares to three different participants.

When a coalition of two participants collaborate together, they can form a matrix B. For example, these two shares are v_1 and v_2 to form the matrix B as

$$B = \begin{bmatrix} 27 & 17 \\ 41 & 21 \\ 76 & 36 \\ 18 & 8 \end{bmatrix}.$$

The projection matrix $\mathbb{S}$ is

$$\mathbb{S} = (B(B'B)^{-1}B') \pmod{19} = \begin{bmatrix} 53 & 87 & 88 & 175 \\ 87 & 137 & 199 & 100 \\ 88 & 199 & 119 & 46 \\ 175 & 100 & 46 & 195 \end{bmatrix}.$$

We can validate that $\mathrm{tr}(\mathbb{S}) = 2 = k$. The secret matrix S is obtained by the remainder matrix R and the projection matrix $\mathbb{S}$ as

$$S = (R + \mathbb{S}) \pmod{19} = \begin{bmatrix} 2 & 3 & 1 & 2 \\ 5 & 4 & 6 & 1 \\ 8 & 9 & 7 & 2 \\ 3 & 4 & 1 & 2 \end{bmatrix}.$$

The reconstructed matrix is the same as the secret matrix, and the shares are $1/m$ of the size of the secret matrix (for our case, it is $1/4$ because $m = 4$). Obviously, this shares compression ratio can be further increased with a larger m value. According to [6, 9], the scheme is not a PSS, rather, this method was proven to be a RSS scheme [2]. Its main advantages are multiple secrets sharing, strong protection of the secrets and smaller size for the secret shares.

3 Proposed Method

Among several interesting properties of matrix projection SSS, an image application can be easily extended from this scheme's ability to share multiple secrets. The pixels in an image can be regarded as elements in a matrix. Although the technique is not a PSS scheme, it has strong protection on the secret [2] even if the remainder matrix R is made public. However, matrix R can become single-point-failure if it is corrupted or lost. To overcome this problem, we propose to use Thien and Lin's method (which is essentially a Shamir's SSS) to share the remainder matrix R without any permutation. As we discussed in section 2, Thien and Lin's method cannot protect matrix R securely, but it does not affect the protection capability on the projection matrix.

For an $l \times l$ secret image with intensity level as $I(i, j)$ where $1 \le i, j \le l$, we can partition the secret image I as non-overlapped $m \times m$ blocks for each RGB color. It procedures roughly $(\lceil l/m \rceil)^2$ blocks. We can share each block S using following scheme

1. construct an $m \times k$ random matrix A of rank k,
2. determine its projection matrix $\mathbb{S}$ and remainder matrix $R = S - \mathbb{S}$.
3. If any element in matrices $\mathbb{S}$ and R is greater than 251, go back to step 1) to reconstruct a new random matrix A. Otherwise, proceed to the next step.
4. Choose n linearly independent $k \times 1$ random vectors x_i and n distinct values r_i,
5. Calculate share $v_i = (A \times x_i) \pmod{p}$ for $1 \le i \le n$.
6. Use Thien and Lin's image SSS to secretly share the matrix R as a $G_i = \left[g_1^{(i)} \quad g_2^{(i)} \quad \ldots \quad g_{\lceil \frac{m}{k} \rceil}^{(i)} \right]$ for $g_t^{(i)}(j) = I(tk+1, j) + \ldots + I(tk+k-1, j) r_t^{k-1} \bmod 251$ where $1 \le t \le \lceil \frac{m}{k} \rceil$ and $1 \le j \le m$
7. Each image share Sh_i is the combination of v_i and G_i.

To illustrate this method in a $(2, 4)$ threshold-based image SSS, we consider a more specific example that a secret matrix is partitioned from a large image where

$$S = \begin{bmatrix} 2 & 3 & 1 & 2 \\ 5 & 4 & 6 & 1 \\ 8 & 9 & 7 & 2 \\ 3 & 4 & 1 & 2 \end{bmatrix}.$$

We see the matrix S is the same secret matrix shown in section 2. Hence, we can use the same secret shares using matrix projection SSS where,

$$v_1 = \begin{bmatrix} 27 \\ 41 \\ 76 \\ 18 \end{bmatrix}, \quad v_2 = \begin{bmatrix} 17 \\ 21 \\ 36 \\ 8 \end{bmatrix}, v_3 = \begin{bmatrix} 11 \\ 9 \\ 12 \\ 2 \end{bmatrix}, \quad v_4 = \begin{bmatrix} 19 \\ 25 \\ 44 \\ 10 \end{bmatrix}, \text{and}$$

the remainder matrix R

$$R = \begin{bmatrix} 51 & 84 & 87 & 173 \\ 82 & 133 & 193 & 99 \\ 80 & 190 & 112 & 44 \\ 172 & 96 & 45 & 193 \end{bmatrix}.$$

Proceedings of the 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC'06)
0-7695-2539-3/06 $20.00 © 2006 IEEE

IEEE COMPUTER SOCIETY

We can set $r_i = i$ for $i = 1, 2, 3, 4$, and

$$G_1 = \begin{bmatrix} 116 & 242 \\ 36 & 210 \\ 232 & 95 \\ 234 & 13 \end{bmatrix}, \quad G_2 = \begin{bmatrix} 32 & 242 \\ 36 & 210 \\ 232 & 95 \\ 234 & 13 \end{bmatrix},$$

$$G_3 = \begin{bmatrix} 199 & 147 \\ 21 & 12 \\ 103 & 7 \\ 42 & 129 \end{bmatrix}, \quad G_4 = \begin{bmatrix} 115 & 225 \\ 139 & 164 \\ 164 & 214 \\ 197 & 187 \end{bmatrix}.$$

Therefore, the image shares are combined as $Sh_i = \begin{bmatrix} v_1 & G_1 \end{bmatrix}$ for $i = 1, 2, 3, 4$, *i.e.*

$$Sh_1 = \begin{bmatrix} 27 & 116 & 242 \\ 41 & 36 & 210 \\ 76 & 232 & 95 \\ 18 & 234 & 13 \end{bmatrix}, \quad Sh_2 = \begin{bmatrix} 17 & 32 & 242 \\ 21 & 36 & 210 \\ 36 & 232 & 95 \\ 8 & 234 & 13 \end{bmatrix},$$

$$Sh_3 = \begin{bmatrix} 11 & 199 & 147 \\ 9 & 21 & 12 \\ 12 & 103 & 7 \\ 2 & 42 & 129 \end{bmatrix}, \quad Sh_4 = \begin{bmatrix} 19 & 115 & 225 \\ 25 & 139 & 164 \\ 44 & 164 & 214 \\ 10 & 197 & 187 \end{bmatrix}.$$

The image shares require $\frac{1}{k} + \frac{1}{m}$ size of the secret image (or $3/4$ size of the secret image because $k = 2$ and $m = 4$). We use the same process for $(2, 4)$ threshold-based image SSS. Four image shares are shown in Figure 2, and

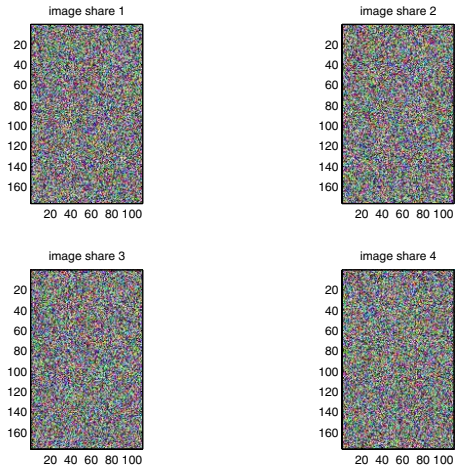

Figure 2. Image Shares for Futurama Image

the reconstructed Futurama image is shown in Figure 3. As we can see from these figures, the size of image shares is significantly less than the size of the secret image. If any image share is corrupted, we can get the secret image back from other three image shares. In fact, any block or blocks of one image share is corrupted, we can still fully recover the whole secret image from corrupted blocks in other three

Figure 3. Lossless reconstructed Futurama Image

image shares. This process does not need to reprocess the whole image as Thien and Lin's method. Also, a secret image can be partitioned into smaller blocks (by setting a smaller value for m) so that the realtime image secret sharing processing can be performed for the whole secret image.

4 Conclusion

We proposed an image SSS using essentially two techniques: i) SSS using matrix projection and ii) Shamir's SSS. A colored secret image can be successfully reconstructed from any k image shares, but cannot be revealed from any $(k-1)$ or fewer image shares (due to RSS scheme for the matrix projection method). The size of image shares is smaller than the size of the secret image. Another advantage is the this scheme can be used in almost realtime by simultaneous processing smaller blocks partitioned from the secret image. For all these block images, we can parallel process the generation of image shares or the reconstruction of the secret image.

References

[1] C. Asmuth and J. Bloom, "A modular approach to key safeguarding," vol. 29, no. 2, pp. 208–210, Mar. 1983.

[2] L. Bai, "A strong ramp secret sharing scheme using matrix projection," presented at the Second International Workshop on Trust, Security and Privacy for Ubiquitous Computing, Niagara-Falls, Buffalo, NY, 2006.

Proceedings of the 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC'06)
0-7695-2539-3/06 $20.00 © 2006 IEEE

IEEE COMPUTER SOCIETY

[3] A. Beimel and B. Chor, "Secret sharing with public reconstruction," vol. 44, no. 5, pp. 1887–1896, Sept. 1998.

[4] G. Blakley, "Safeguarding crypographic keys," presented at the Proceedings of the AFIPS 1979 National Computer Conference, vol. 48, Arlington, VA, June 1997, pp. 313–317.

[5] G. R. Blakley and C. Meadows, "Security of ramp schemes," presented at the Advances in Cryptology – Crypto '84, G. R. Blakley and D. Chaum, Eds., Aug. 1984.

[6] R. M. Capocelli, A. De Santis, L. Gargano, and U. Vaccaro, "On the size of shares for secret sharing schemes," presented at the Advances in Cryptology – Crypto '91, J. Feigenbaum, Ed., vol. 576, Santa Barbara, California, USA, Aug. 1991, pp. 101–113.

[7] A. De Santis and B. Masucci, "Multiple ramp schemes," vol. 45, no. 5, pp. 1720–1728, July 1999.

[8] M. Franklin and M. Yung, "Communication complexity of secure computation," 1992. [Online]. Available: citeseer.ifi.unizh.ch/franklin92communication.html

[9] E. D. Karnin, J. W. Greene, and M. E. Hellman, "On secret sharing systems," vol. IT-29, no. 1, pp. 35–41, Jan. 1983.

[10] K. Kurosawa, K. Okada, K. Sakano, W. Ogata, and S. Tsujii, "Nonperfect secret sharing schemes and matroids," *Lecture Notes in Computer Science*, vol. 765, pp. 126–141, 1994. [Online]. Available: citeseer.ist.psu.edu/article/kurosawa94nonperfect.html

[11] T. Migler, K. E. Morrison, and M. Ogle, "Weight and rank of matrices over finite fields," September 29 2003. [Online]. Available: http://www.calpoly.edu/ kmorriso/Research/weight.pdf

[12] M. Naor and A. Shair. (1996, June) Visual cryptography II: Improving the contrast via the cover base. [Online]. Available: http://philby.ucsd.edu/cryptolib/1996/96-07.html. Last accessed: March 2005.

[13] M. Naor and A. Shamir, "Visual cryptography," presented at the Proceedings of the Conference on Advances in Cryptology – Eurocrypt '94, A. De Santis, Ed., Berlin, Germany, 1994, pp. 1–12.

[14] W. Ogata and K. Kurosawa, "Some basic properties of general nonperfect secret sharing schemes," *J.UCS: Journal of Universal Computer Science*, vol. 4, no. 8, pp. 690–704, 1998. [Online]. Available: citeseer.ist.psu.edu/article/ogata98some.html

[15] P. Paillier, "On ideal non-perfect secret sharing schemes," in *Security Protocols Workshop*, 1997, pp. 207–216. [Online]. Available: citeseer.ist.psu.edu/paillier98ideal.html

[16] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979.

[17] K. Srinathan, N. Tharani Rajan, and C. Pandu Rangan, "Non-perfect secret sharing over general access structures," in *INDOCRYPT*, 2002, pp. 409–421.

[18] C.-C. Thien and J.-C. Lin, "Secret image sharing," *Computers & Graphics*, vol. 26, no. 5, pp. 765–770, 2002.

[19] 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. 1277–1385, 2004.

Proceedings of the 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC'06)
0-7695-2539-3/06 $20.00 © 2006 IEEE

IEEE COMPUTER SOCIETY