Weighted Threshold Secret Image Sharing*

Shyong Jian Shyu[1,**], Chun-Chieh Chuang[2], Ying-Ru Chen[1], and Ah-Fur Lai[2]

[1] Department of Computer Science and Information Engineering, Ming Chuan University, No. 5, De-Ming Road, Guei-Shan, Taoyuan 33348, Taiwan
Tel.: 886-3-3507001 Ext.3404; Fax: 886-3-3593874
[2] Department of Computer Science, Taipei Municipal University of Education, No. 1, Ai-Guo West Road, Taipei 10048, Taiwan
sjshyu@mail.mcu.edu.tw, {hellion0724,cindyooxx}@gmail.com, lai@tmue.edu.tw

Abstract. Given a secret image I, a threshold r, and a set of n ($\geq r$) participants $\mathcal{P} = \{1, 2, \ldots, n\}$ with a set of weights $\mathcal{W} = \{w_1, w_2, \ldots, w_n\}$ where w_i is the weight (which indicates the degree/rank of importance) of participant i and we assume that $w_1 \leq w_2 \leq \ldots \leq w_n$. The idea of weighted threshold secret image sharing encodes I into n shadows $S_1, S_2, \ldots, S_n$ with sizes $|S_1| \leq |S_2| \leq \ldots \leq |S_n|$ in which S_i is distributed to participant i such that only when a group of r participants can reconstruct I by using their shadows, while any group of less than r participants cannot. We propose a novel weighted threshold secret image sharing scheme based upon Chinese remainder theorem in this paper. As compared to the conventional Shamir's and recent Thien-Lin's schemes, which produce shadows with the same size, our scheme is more flexible due to the reason that the dealer is able to distribute various-sized shadows to participants with different degrees/ranks of importance in terms of practical concerns.

Keywords: Threshold secret sharing, Secret image sharing, Weighted shadow size, Chinese remainder theorem.

1 Introduction

The concept of *threshold secret sharing* aims at sharing a secret value among several participants where each participant owns a part of the secret called *shadow* and only when a certain number (called *threshold*) of participants utilize their shadows can the secret be reconstructed, while less than the threshold number of participants cannot. Consider a secret s and a set of participants $P = \{1, 2, \ldots, n\}$ sharing s. Any approach that achieves the requirements of secret sharing for s with a threshold r among the n participants in P is called an r out of n (or (r, n)) secret sharing scheme.

Shamir [14] and Blakley [2] independently proposed the threshold secret sharing schemes in 1979. Shamir's approach is based upon the *polynomial interpolation* in a

* This research was partly supported by National Science Council of the Republic of China under contract NSC-97-2221-E-130-022-MY3.
** Corresponding author.

T. Wada, F. Huang, and S. Lin (Eds.): PSIVT 2009, LNCS 5414, pp. 988–998, 2008.

two-dimensional space, while Blakley's scheme originates from the intersections of some high-dimensional planes in a high-dimensional space. Shamir's scheme is simple and easy to implement so that it has attracted many researchers' attention [3, 4, 9, 11, 16]. Consider an $r-1$ degree polynomial:

$$f(x) = a_0 + a_1x^1 + a_2x^2 + \ldots + a_{r-1}x^{r-1} \quad (1)$$

where all computations are perform in $GF(p)$ in which p is a prime (or a power of 2 or a prime), $1 \le a_{r-1} < p$, $0 \le a_w < p$ for $0 \le w \le r-2$, and $1 \le x < p$. Shamir's (r, n) scheme utilizes this polynomial to share a secret s as follows. The dealer sets s to be a_0 and randomly chooses $a_1, a_2, \ldots, a_{r-1}$ to form $f(x)$. Then, he/she chooses $x_1, x_2, \ldots, x_n$ as *keys* based upon which $f(x_1), f(x_2), \ldots, f(x_n)$ are computed as *shadows*. The n pairs of $(f(x_i), x_i)$'s, $1 \le i \le n$, are distributed to the n participants one by one. Since any group of r (or more) $(f(x_i), x_i)$'s is able to compute $(a_0, a_1, \ldots, a_{r-1})$ by solving the r equations by polynomial interpolation, s $(= a_0)$ is thus recovered. None of any group of less than r participants can solve the r equations completely. We say that s is shared by n participants in an (r, n) threshold structure.

Thien and Lin [17] in 2002 extended Shamir's scheme so that the idea can be apply to share a secret image. Consider an image P with N pixels in total to be shared in an (r, n) threshold structure. Thien-Lin's scheme first diffuses all N pixels in P and organizes them into N/r segments with r pixels each. Let the r pixels in segment t be denoted as $(a_0, a_1, \ldots, a_{r-1})_t$, $1 \le t \le N/r$. The values of these r pixels of segment t are assigned to be the r coefficients of the above formula to form $f_t(x)$. Then, the dealer determines n keys $x_1, x_2, \ldots, x_n$, and computes $f_t(x_1), f_t(x_2), \ldots, f_t(x_n)$ for $1 \le t \le N/r$. After that, $f_1(x_i), f_2(x_i), \ldots, f_{N/r}(x_i)$ are merged into a *shadow image* D_i for $1 \le i \le n$. The dealer gives (D_i, x_i) to participant i for $1 \le i \le n$. It is not hard to see that only r (or more) participants can recover $(a_0, a_1, \ldots, a_{r-1})_t$ by using their r pairs of keys and shadows in polynomial interpolation for all equations $f_t(x)$'s, $1 \le t \le N/r$. $(a_0, a_1, \ldots, a_{r-1})_1$, $(a_0, a_1, \ldots, a_{r-1})_2$, $\ldots$, $(a_0, a_1, \ldots, a_{r-1})_{N/r}$ are indeed the N pixels in P which have been diffused before. After re-order all of the pixels, we reconstruct P. The shadow size of Thien-Lin's approach is N/r, that is, D_i contains N/r pixels for $1 \le i \le n$. If the original Shamir's approach is directly applied to share an image, the size of each shadow is N. Therefore, Thien-Lin's scheme reduces the size of the shadows as compared to Shamir's.

However, the sizes of all shadow images are the same in either Thien-Lin's or Shamir's approach. In real-world applications, this might not always be an advantage. For instance, a particular participant (the boss, a secret agent, etc.) would like to carry a shadow with a smaller (or larger) size for reducing the burden, cost (or increasing the secrecy) or other reasons. Our interest in this paper is thus to design a threshold secret image sharing scheme which produces shadows with various sizes. Since the dealer could define the weights of the participants and distribute the different-sized shadows to the participants according to their weights, we call our design the *weighted threshold secret image sharing scheme*. Essentially, the proposed scheme is based upon the *Chinese remainder theorem*.

The rest of the paper is organized as follows. We introduce Chinese remainder theorem and how to apply CRT to accomplish secret sharing in Section 2. Our design for a weighted threshold secret image sharing scheme is proposed in Section 3. Some

experiments results are reported in Section 4. The secrecy analysis of our scheme is discussed in Section 5. Section 6 gives some concluding remarks.

2 Previous Studies

2.1 Chinese Reminder Theorem

Consider a secret value x and $m \geq 2$ positive relatively prime moduli, namely $q_1, q_2, \ldots, q_m$. Let $Q = q_1 \times q_2 \times \ldots \times q_m$ and s_i be the remainder of x modulo q_i for $1 \leq i \leq m$. The Chinese remainder theorem (CRT) asserts that the following system has a unique solution x in Z_Q [6]:

$$\begin{aligned} x &\equiv s_1 \pmod{q_1} \\ x &\equiv s_2 \pmod{q_2} \\ &\cdots \\ x &\equiv s_m \pmod{q_m} . \end{aligned} \tag{2}$$

Give a number x and m positive relatively prime moduli $q_1, q_2, \ldots, q_m$ where $x \in Z_Q$, the above system is described as:

$$(s_1, s_2, \ldots, s_m) = CRT\_remainders(x, m, q_1, q_2, \ldots, q_m). \tag{3}$$

The solution x in Z_Q can be obtained by many ways. One of the popular approaches is to compute M_i and its multiple inverse c_i (under modulus q_i) for all moduli q_i, $1 \leq i \leq m$ [15] first as follows:

$$M_i = Q / q_i , \tag{4}$$

$$c_i M_i \equiv 1 \bmod q_i . \tag{5}$$

Then x can be obtained by

$$x = \left(\sum_{i=1}^{m} s_i c_i M_i \right) \bmod Q . \tag{6}$$

To ease the following applications of finding a solution based upon CRT, we organize these operations a procedure:

$$x = CRT\_solution(m, q_1, q_2, \ldots, q_m, s_1, s_2, \ldots, s_m) \tag{7}$$

where $x \equiv s_i \pmod{q_i}$ for $1 \leq i \leq m$.

2.2 Threshold Secret Sharing by CRT

Let x be a secret value and $q_1, q_2, \ldots, q_m$ be m positive relatively prime moduli where $Q = q_1 \times q_2 \times \ldots \times q_m$ and $x \in Z_Q$. Since $(s_1, s_2, \ldots, s_m) = CRT\_remainder(x, m, q_1, q_2, \ldots, q_m)$, a naïve idea for applying CRT for sharing x among m participants may be using s_i as the shadow for participant i, $1 \leq i \leq m$. (This was adopted by Meher and Patra [12] in their secret image sharing scheme in 2006.) For instance, assume that $m = 3$ and $(q_1, q_2,$

q_3) = (3, 5, 7). Consider a secret $x = 97$ sharing by 3 (= m) participants. Since (s_1, s_2, s_3) = (1, 2, 6) (= *CRT_remainder*(97, 3, 3, 5, 7)), i.e.

$$97 \equiv 1 \bmod 3$$
$$97 \equiv 2 \bmod 5$$
$$97 \equiv 6 \bmod 7 ,$$

(s_i, q_i) might be distributed to participant i for $i = 1, 2, 3$. Then, only when all three participants utilize their information, they can compute $x = 97$; while any group of less than two participants cannot.

Yet, we give an example to illustrate that such naïve application is incorrect in some cases. Consider the same scenario except for $x = 18$. We have (s_1, s_2, s_3) = (0, 3, 4) (= *CRT_remainder*(18, 3, 3, 5, 7)):

$$18 \equiv 0 \bmod 3$$
$$18 \equiv 3 \bmod 5$$
$$18 \equiv 4 \bmod 7 .$$

Indeed, all three participants can obtain 18 (18 = *CRT_solution*(3, 3, 5, 7, 0, 3, 4). However, participants 1 and 3 (or 2 and 3) can do so by using their (0, 3) and (4, 7) (or (3, 5) and (4, 7)) (18 = *CRT_solution*(2, 3, 7, 0, 4) = *CRT_solution*(2, 5, 7, 3, 4)) too. Thus, it is not a (3, 3) scheme. This naïve application of CRT cannot construct a threshold secret sharing scheme.

To share a secret by using CRT is not a new topic, Mignotte [13] and Asmuth-Bloom [1] proposed (r, n) threshold secret sharing schemes in 1983 individually. Some following studies can be found in [5, 7, 8, 10]. Our scheme is based upon Mignotte's idea that is introduced as follows.

Consider n relatively positive prime moduli $q_1 < q_2 < \ldots < q_n$. Let $\alpha = q_{n-r+2} \times q_{n-r+3} \times \ldots \times q_n$ (the product of maximal $r-1$ moduli) and $\beta = q_1 \times q_2 \times \ldots \times q_r$ (the product of the minimal r moduli). Let secret x satisfy $\alpha < x < \beta$. The dealer distributes (s_i, q_i) to participant i for $1 \le i \le n$ where (s_1, s_2, … , s_n) = *CRT_remainder*(x, n, q_1, q_2, … , q_n) so as to accomplish sharing x among the n participants in an (r, n) structure. Assume that any group of $r-1$ participants, say $\{i_1, i_2, \ldots, i_{r-1}\}$, compute as follows with their shadows and moduli:

$$y = CRT\_solution(r-1, q_{i_1}, q_{i_2}, \ldots, q_{i_{r-1}}, s_{i_1}, s_{i_2}, \ldots, s_{i_{r-1}}) .$$

They can only retain a solution y in $Z_{Q'}$ where $Q' = q_{i_1} \times q_{i_2} \times \ldots \times q_{i_{r-1}} \le \alpha (= q_{n-r+2} \times q_{n-r+3} \times \ldots \times q_n)$ according to CRT. Since $y < \alpha < x$, $y \ne x$. On the other hand, when r participants, say $i_1, i_2, \ldots, i_r$, compute as follows with all their shadows and moduli, they can recover x:

$$x = CRT\_solution(r, q_{i_1}, q_{i_2}, \ldots, q_{i_r}, s_{i_1}, s_{i_2}, \ldots, s_{i_r}) .$$

Therefore, the (r, n) threshold property holds.

3 The Proposed Scheme

Consider an $h \times w$ secret image I with M bits in total and a set of n participants sharing I. Our encoding process first chooses n relatively prime moduli $q_1 < q_2 < \ldots < q_n$, and compute $\alpha = q_{n-r+2} \times q_{n-r+3} \times \ldots \times q_n$ and $\beta = q_1 \times q_2 \times \ldots \times q_r$. We regard secret image I as a series of l blocks with d-bit each (i.e. $d \times l = M$) block where each block, say I_k, as an encoding unit where $0 \le x_k = value(I_k) \le 2^d - 1$ for $1 \le k \le l$ where $value(I_k)$ denotes the d-bit value of I_k. To cope with cases like natural images which contains similar colors (values of pixels), we simply introduce a series of random numbers in range $[0, 2^d - 1]$ with an initial seed e and perform "xor" operation to all values of all blocks in order to diffuse those similar colors.

To maintain the (r, n) threshold property, we adjust the value of x_k to be x_k' such that the constraint $\alpha < x_k' < \beta$ is met. This is done by adding the diffused value x_k' with a pre-determined offset p where $\alpha < p < \beta - 2^d$. Formally, we set e as the seed of the random sequence, i.e.

$$random\_seed(e) \tag{8}$$

and set the range of the random numbers, i.e. $[0, 2^d - 1]$, by

$$random\_range(0{:}2^d - 1)\ ; \tag{9}$$

then perform

$$x_k' = (x_k \oplus random()) + p \tag{10}$$

for all I_k's, $1 \le k \le l$ where $random()$ returns a random number which is a member of a random sequence in $[0, 2^d - 1]$ seeded by e. Note that we deliberately set p as the seed e, i.e. $e = p$ in our implementation. Then, x_k' is shared among the n participants in an (r, n) structure by using CRT for all I_k's:

$$(s_{k,1}, s_{k,2}, \ldots, s_{k,n}) = CRT\_remainder(x_k', n, q_1, q_2, \ldots, q_n). \tag{11}$$

where $0 \le s_{k,i} < q_i$. We take $z_i = \lceil \log_2 q_i \rceil$ bits to store $s_{k,i}$ for $1 \le k \le l$ and $1 \le i \le n$. All z_i-bit remainders distributed to participant i are merged z_i-bit by z_i-bits to form shadow S_i:

$$S_i = s_{1,i} \| s_{2,i} \| \ldots \| s_{l,i} \tag{12}$$

where $\|$ denotes the concatenation operation. Thus, the bit-length of S_i is $|S_i| = z_i \times l$ $(= \lceil \log_2 q_i \rceil \times M/d)$.

Further, p is shared among the n participants in the (r, n) structure by using CRT, too. That is

$$(a_1, a_2, \ldots, a_n) = CRT\_remainder(p, n, q_1, q_2, \ldots, q_n)\ . \tag{13}$$

The dealer thus distributes (S_i, a_i, q_i) to participant i for $1 \le i \le n$. Since $q_1 < q_2 < \ldots < q_n$, we have $(\lceil \log_2 q_1 \rceil \times M/d) \le (\lceil \log_2 q_2 \rceil \times M/d) \le \ldots \le (\lceil \log_2 q_n \rceil \times M/d)$ and consequently $|S_1| \le |S_2| \le \ldots \le |S_n|$. That means the sizes of the shadows are weighted in terms of those of the moduli. Or, each participant receives a part of information whose size is related to his/her weight.

The encoding algorithm is formally illustrated as follows.

Encoding algorithm

Input: a secret image I with M bits in total, a set of participants $P = \{1, 2, \ldots, n\}$ with a set of weights $W = \{w_1, w_2, \ldots, w_n \mid w_1 \le w_2 \le \ldots \le w_n\}$, threshold r ($2 \le r \le n$), and parameter d.

Output: shadows S_i and a_i, and modulus q_i for $1 \le i \le n$.

1. Choose $\{q_1, q_2, \ldots, q_n \mid (q_i, q_j) = 1, 2 < q_1 < q_2 < \ldots < q_n < 2^d\}$ according to W and d
2. $\alpha = q_{n-r+2} \times q_{n-r+3} \times \ldots \times q_n$; $\beta = q_1 \times q_2 \times \ldots \times q_r$ // $\alpha < \beta < 2^d$
3. Choose seed p randomly with $\alpha < p < \beta - 2^d$
4. *random_seed*(p); *random_range*($0{:}2^d - 1$)
   // set p as the seed of the random sequence ranging from 0 to 2^d
5. Partition I into l ($= M/d$) segments: $I_1, I_2, \ldots, I_l$ // I_k is with d bits, $1 \le k \le l$
6. for (each I_k, $1 \le k \le l$) do
6.1 { $x_k = value(I_k)$
6.2 $x_k' = (x_k \oplus random()) + p$
6.3 for (each i, $1 \le i \le n$) do $s_{k,i} = x_k' \bmod q_i$ // $|s_{k,i}| = \lceil \log_2 q_i \rceil$
}
7. for (each i, $1 \le i \le n$) do
7.1 { $S_i = \varnothing$
7.2 for (each k, $1 \le k \le l$) do $S_i = S_i \cup \{s_{k,i}\}$
// Append $s_{k,i}$ ($|s_{k,i}| = \lceil \log_2 q_i \rceil$) after S_i: $S_i = S_i \| s_{k,i}$
}
8. for (each i, $1 \le i \le n$) do $a_i = p \bmod q_i$
9. Output($S_1, S_2, \ldots, S_n, a_1, a_2, \ldots, a_n, q_1, q_2, \ldots, q_n$)
   // the dealer distributes (S_i, a_i, q_i) to participant i

Participant i would get (S_i, a_i, q_i) from the dealer for $1 \le i \le n$. It is noticed that the size of shadow S_i is $\lceil \log_2 q_i \rceil \times M/d$ for $1 \le i \le n$. Thus the sizes of $S_1, S_2, \ldots, S_n$ are determined by those of $q_1, q_2, \ldots, q_n$ which are based upon the weights $w_1, w_2, \ldots, w_n$ accordingly. This offers a flexible decision for the dealer about which participant is more/less important at his/her convenience.

The decoding algorithm is shown in the following.

Decoding algorithm

Input: r participants $i_1, i_2, \ldots, i_r \in P$ and the corresponding moduli $q_{i_1} < q_{i_2} < \ldots < q_{i_r}$, shadows $S_{i_1}, S_{i_2}, \ldots, S_{i_r}$, and $a_{i_1}, a_{i_2}, \ldots, a_{i_r}$, and parameter d.

Output: the secret image I.

1. $p = CRT\_solution(r, a_{i_1}, a_{i_2}, \ldots, a_{i_r}, q_{i_1}, q_{i_2}, \ldots, q_{i_r})$
2. for ($1 \le j \le r$) $z_j = \lceil \log_2 q_{i_j} \rceil$
3. *random_seed*(p); *random_range*($0{:}2^d - 1$)
4. $I = \varnothing$
5. $l = |S_{i_1}| / z_1$ // l is the number of blocks; each shadow has the same l

6. for (each k, $1 \le k \le l$) do
6.1 { for (each S_{i_j}, $1 \le j \le r$) do
{ $s_{k,j}$ = the first z_j bits of S_{i_j}
$S_{i_j} = S_{i_j} - \{s_{k,j}\}$ // delete the first z_j bits from S_{i_j}
}
6.2 $y_k = CRT\_solution(r, s_{k,1}, s_{k,2}, \ldots, s_{k,r}, q_{i_1}, q_{i_2}, \ldots, q_{i_r})$
6.3 $x_k = (y_k - p) \oplus random()$
6.4 make x_k to be d-bit long
6.5 $I = I \cup \{x_k\}$ // Append x_k after I by d-bit concatenation ($I = I \| x_k$)
}
7. Output(I)

4 Experimental Results

We report the implementation results of our scheme for testing a simple (3, 4) case in this section. The program was coded in Microsoft C# and tested in a PC with Windows. A 256×256 color Mandrill image was tested the secret image I as shown in Fig. 1 which is shared by four participants: 1, 2, 3 and 4 with weights $w_1 \le w_2 \le w_3 \le w_4$. We assume that the dealer would like to produce four shadows S_1, S_2, S_3 and S_4 for participants 1, 2, 3 and 4 respectively with $|S_1| \le |S_2| \le |S_3| \le |S_4|$.

In our implementation, we set d as 29 and (q_1, q_2, q_3, q_4) = (1009, 2026, 5095, 31651); thus, α = 5095×31651 = 161261845 and β = 1009×2026×5095 = 10415372230. The secret image is treated as a one dimensional array with 256×256×24 bit (since one color pixel takes 24 bits specifying the R, G, B colors in a Windows environment). The number of blocks in our experiment is $l = \lceil M/d \rceil = 54237$. Note that we simply append white pixels in the last block to make the number of pixels in it to be 29.

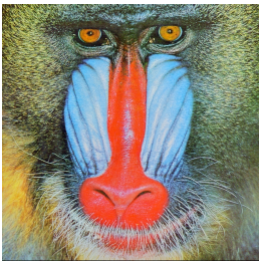
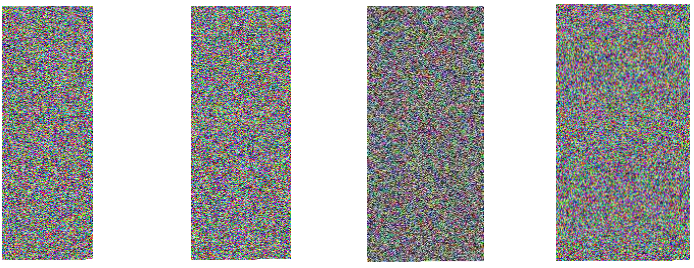


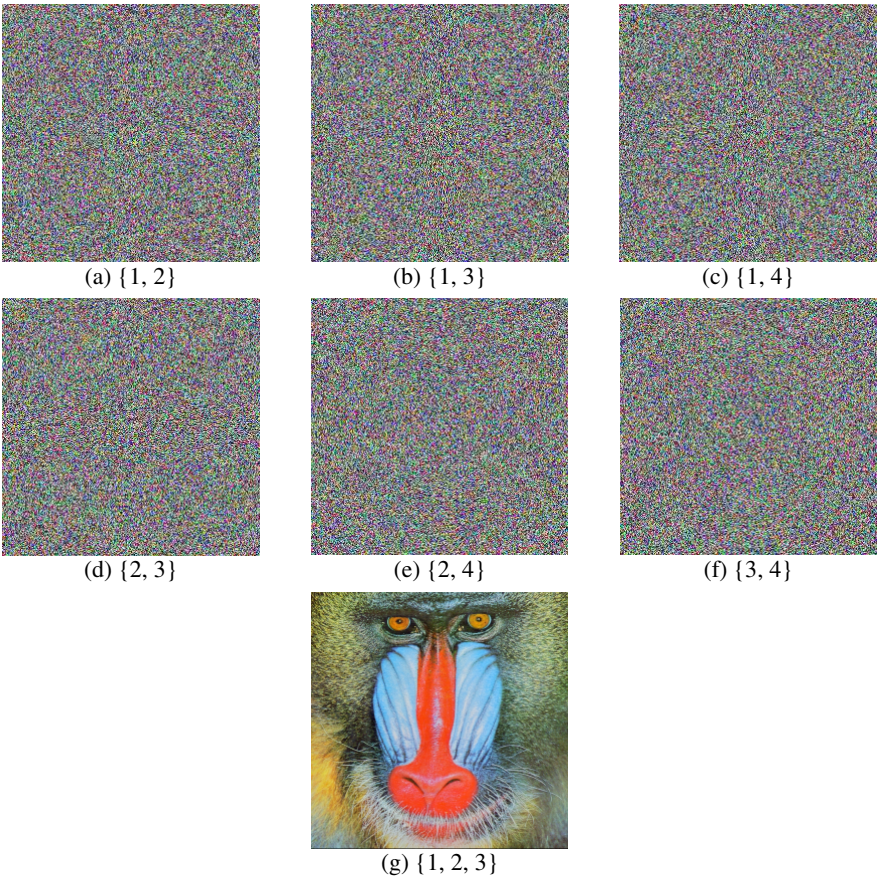
Fig. 1. Secret image to be shared

Fig. 2 shows the four shares S_1, S_2, S_3 and S_4 produced by our encoding algorithm with sizes 89×256, 98×256, 115×256 and 133×256 respectively which meet the requirement of $|S_1| \le |S_2| \le |S_3| \le |S_4|$. Let us explain why the size (or pixels) of S_1 is 89×256. Each remainder of the value corresponding to a 29-bit block in I under modulus q_1 (= 1009) is less than 1009 and is stored by using $\lceil \log_2 q_1 \rceil = \lceil \log_2 1009 \rceil = 10$ bits.



(a) S_1 (b) S_2 (c) S_3 (d) S_4

Fig. 2. Shadows produced by the encoding algorithm



(a) {1, 2} (b) {1, 3} (c) {1, 4}

(d) {2, 3} (e) {2, 4} (f) {3, 4}

(g) {1, 2, 3}

Fig. 3. Reconstructed results from the decoding algorithm by various groups of participants

Thus, after encoding all l blocks, there are 54237×10 = 542370 encoded bits which constitute S_1. The bit-lengths of the other shadows are determined in the same way. For the display and comparison purposes, we regarded these consecutive bits as a serious of 24-bit color pixels which constitute a color image with a height of 256. Since $\lceil (542370/24)/256 \rceil = 89$, thus the size of S_1 becomes 89×256.

Fig. 3 illustrates the reconstructed images from our decoding algorithm by various groups of participants where (a)-(g) are by {1, 2}, {1, 3}, {1, 4}, {2, 3}, {2, 4}, {3, 4}, {1, 2, 3} respectively. Note that the results obtained by {1, 2, 4}, {1, 3, 4}. {2, 3, 4} and {1, 2, 3, 4} are exactly the same as Fig. 3 (g), which is the same as the original baboon image, so that we just omit them here. Besides, the sizes (or pixels) of these resultant images are all 256×256. This is due to our assumption that the groups of more than one participant knew d (the block size), l (the number of blocks) and the decoding algorithm so that they applied CRT to recover the 29-bit secret blocks by using their information and displayed their result as a 24-bit based color image.

It is easily seen from Fig. 3 that any group of less than three participants cannot recover I, while any group of three or more participants can. The attractive feature is that $|S_1| \le |S_2| \le |S_3| \le |S_4|$ whose sizes are determined by the values of the chosen moduli which define the degree/rank of importance of the participants. These results demonstrated the feasibility and applicability of our scheme.

5 Secrecy Analysis

Since p is shared in an (r, n) manner, thus, only when r participants can recover them correctly. Assume that a group of $r-1$ participants, say $\{i_1, i_2, \ldots, i_{r-1}\}$ with $w_{i_1} < w_{i_2} < \ldots < w_{i_{r-1}}$, tries to recover I by using information what they have. The best case for them is to guess i_0's information $(S_{i_0}, a_{i_0}, q_{i_0})$ where $i_0 < i_1$ (or $q_{i_0} < q_{i_1}$) (since it would be harder to guess i_j's ($> i_{r-1}$) information, $j > r-1$). Due to the reason that the number of blocks l is the same among all participants, we have $l = |S_{i_j}| / \lceil \log_2 q_{i_j} \rceil$ for each j, $1 \le j \le r-1$. That is, we know the number of blocks in S_{i_0}. Still, a_{i_0}, q_{i_0} and the content of S_{i_0} are unknown by now. Since a_{i_0} is a certain remainder under modulus q_{i_0}, thus $a_{i_0} < q_{i_0}$. The probability for blindly guessing a_{i_0} and q_{i_0} right is $1/(a_{i_0})\times 1/(q_{i_0})$ $(< 1/(q_{i_1})^2$ because of $a_{i_0} < q_{i_0} < q_{i_1})$. Once a_{i_0} and q_{i_0} are correctly obtained, p can be obtained by computing

$$p = CRT\_solution(r, a_{i_0}, a_{i_1}, \ldots, a_{i_{r-1}}, q_{i_0}, q_{i_1}, \ldots, q_{i_{r-1}}) .$$

By guessing the value $s_{k,0}$ in block k of $\lceil \log_2 q_{i_0} \rceil$ bits in S_{i_0} (i.e. a remainder under modulus q_{i_0}), we compute:

$$x_k' = CRT\_solution(r, s_{k,0}, s_{k,1}, \ldots, s_{k,r-1}, q_{i_0}, q_{i_1}, \ldots, q_{i_{r-1}})$$

where $s_{k,j}$ is the value of block k in S_{i_j} for $0 \le j \le r-1$ and $1 \le k \le l$. By using

$$x_k = (x_k' - p) \oplus random() ,$$

we get x_k, which is the value of block I_k with d bits in I where *random*() returns a random number seeded initially at p with a range of $[0, 2^d-1]$. If all x_k's are found and merged (d-bit by d-bit) for $1 \le k \le l$, we obtain I. That is $I = x_1 \| x_2 \| \ldots \| x_l$.

The probability of guessing $s_{k,0}$ right (and subsequently finding out x_k) is $1/q_{i_0}$ ($< 1/q_{i_1}$) and a bound of that of guessing all l blocks' $s_{k,0}$'s ($1 \le k \le l$) right (and subsequently finding out I) is $(1/q_{i_1})^l$. Thus an upper bound of the probability for this group of $r-1$ participants to decode $(S_{i_0}, a_{i_0}, q_{i_0})$ is

$$(1/(q_{i_1})^2)\times(1/q_{i_1})^l = (1/q_{i_1})^{l+2} \; (= (1/q_{i_1})^{M/d+2}) \; . \tag{14}$$

As mentioned earlier, once $(S_{i_0}, a_{i_0}, q_{i_0})$ is found, the secret image I can be reconstructed by this group of the $r-1$ participants.

6 Concluding Remarks

We propose, analyze and implement a novel weighted threshold secret image sharing scheme by using CRT in this paper. The shadow sizes produced by our scheme are correlated with the weights of the participants which imply the degrees/ranks of importance of the participants. As compared to the conventional Shamir's and the recent Thien-Lin's approaches which produce shadows with the same size, our scheme is more flexible so that it can be applied to some practical situations that the parts of information given to different participants are with different sizes in terms of their degrees/ranks of importance.

In the decoding and encoding algorithms, d is designed to be an input parameter and the seed e is the same as p. To increase the level of secrecy, d and e might be shared as well among the n participants in an (r, n) structure. For a certain parameter d, how to determine the n relative prime moduli (all less than 2^d) is a critical concern in our scheme, especially, to enhance the level of secrecy (as discussed in Section 5). It is an interesting topic to go on. The authors are also interested in how to find the n relative prime moduli which are proportional to the values of the weights efficiently.

References

1. Asmuth, C., Bloom, J.: A modular approach to key safeguarding. IEEE Transactions on Information Theory IT-29(2), 208–210 (1983)
2. Blakley, G.R.: Safeguarding cryptographic keys. In: AFIPS Conf. Proc., vol. 48, pp. 313–317 (1979)
3. Brickell, E.F.: Some ideal secret sharing schemes. Journal of Combinatorial Mathematics and Combinatorial Computing 6, 105–113 (1989)
4. Chang, C.-C., Hwang, R.-J.: Sharing secret images using shadow codebooks. Information Sciences 111(1-4), 335–345 (1998)
5. Galibus, T., Matveev, G.: Generalized mignotte's sequences over polynomial rings. Electr. Notes Theor. Comput. Sci. 186, 43–48 (2007)
6. Hardy, D.W., Walker, C.L.: Applied Algebra: codes, ciphers, and discrete algorithms. Prentice Hall, Englewood Cliffs (2003)

7. Iftene, S.: Compartmented secret sharing based on the Chinese remainder theorem. Cryptology ePrint Archive (2005)
8. Iftene, S.: General secret sharing based on the Chinese remainder theorem. Cryptology ePrint Archive, Report 2006/166 (2006)
9. Ito, M., Saito, A., Nishizeki, T.: Secret sharing scheme realizing general access structure. In: Proceedings of IEEE, Globecom 1987, pp. 99–102 (1987)
10. Li, H.-X., Pang, L.-J., Cai, W.-D.: An efficient threshold multi-group-secret sharing scheme. Advances in Soft Computing 40, 911–918 (2007)
11. Lin, C.-C., Tsai, W.-H.: Secret image sharing with steganography and authentication. Journal of Systems and Software 73(3), 405–414 (2004)
12. Meher, P.K., Patra, J.C.: A new approach to secure distributed storage, sharing and dissemination of digital image. In: Proceedings of the IEEE International Symposium on Circuits and Systems, pp. 373–376 (2006)
13. Mignotte, M.: How to share a secret. In: Beth, T. (ed.) EUROCRYPT 1982. LNCS, vol. 149, pp. 371–375. Springer, Heidelberg (1983)
14. Shamir, A.: How to share a secret. Communications of the ACM 22(11), 612–613 (1979)
15. Stallings, W.: Cryptography and Network Security Principles and Practices, 4th edn. Prentice Hall, Englewood Cliffs (2005)
16. Tan, K.J., Zhu, H.W.: General secret sharing scheme. Computer Communications 22, 755–757 (1999)
17. Thien, C.-C., Lin, J.-C.: Secret image sharing. Computers and Graphics 26, 765–770 (2002)