Journal of Electronic Imaging

SPIEDigitalLibrary.org/jei

# Tamper-proof secret image-sharing scheme for identifying cheated secret keys and shared images

Chien-Chang Chen
Chong-An Liu

SPIE

IS&T imaging.org

# Tamper-proof secret image-sharing scheme for identifying cheated secret keys and shared images

**Chien-Chang Chen**
**Chong-An Liu**
Tamkang University
Department of Computer Science and Information Engineering
Taipei, Taiwan
E-mail: ccchen34@mail.tku.edu.tw

**Abstract.** *A $(t, n)$ secret image-sharing scheme shares a secret image to n participants, and the t users recover the image. During the recovery procedure of a conventional secret image-sharing scheme, cheaters may use counterfeit secret keys or modified shared images to cheat other users' secret keys and shared images. A cheated secret key or shared image leads to an incorrect secret image. Unfortunately, the cheater cannot be identified. We present an exponent and modulus-based scheme to provide a tamper-proof secret image-sharing scheme for identifying cheaters on secret keys or shared images. The proposed scheme allows users to securely select their secret key. This assignment can be performed over networks. Modulus results of each shared image is calculated to recognize cheaters of a shared image. Experimental results indicate that the proposed scheme is excellent at identifying cheated secret keys and shared images. © 2013 SPIE and IS&T.* [DOI: 10.1117/1.JEI.22.1.013008]

## 1 Introduction

Sharing images secretly is essential to protect important images. Conventional $(t, n)$ secret image-sharing methods share one secret image to $n$ shared images, and gathering $t$ shared images recovers the secret image. Thien and Lin[1] presented an efficient secret image-sharing scheme by using Shamir method for image sharing and using the Lagrange interpolation method for reconstruction. Many researchers further present functional image-sharing ideas (e.g., reducing load in sharing multiple images,[2] progressive,[3–6] weighted,[7] visual cryptography and secret image sharing,[8,9] scalable,[10] and sharing with hiding).[11] In addition to the Shamir-Lagrange method, many other methods such as Blakley,[12] Boolean,[13] and Chinese Remainder Theorem[14] are also adopted to share important images secretly.

Although numerous secret image-sharing methods have been proposed, an efficient method of detecting cheaters both in secret key and shared image has not been presented. Currently, the convenience of computer networks allows users to share and recover a secret image over networks easily. However, hackers may use counterfeit secret keys or modified shared image to misappropriate other participants' authorized secret keys and shared images. Therefore a structure of applying tamper-proof secret image-sharing techniques over computer networks merits the current study. Research has been presented to introduce a method for identifying cheaters. Wu and Wu[15] used hash functions to collect shared messages and then generated a large number for verification. Chang and Hwang[16] improved the Wu and Wu scheme to increase security by factoring the product of two large prime numbers. Tan et al.[17] presented a quadratic residue-based secret sharing scheme. Other researchers further discussed cheaters' identification approaches to secret image-sharing problems. Chen and Suen[18] adopted the Wu and Wu scheme to verify the authenticity of shared images. Zhao et al.[19] presented an exponent computation-based secret key verification scheme. Although some works on detecting cheaters in secret image-sharing problems are present, a complete solution for detecting cheaters both in secret keys and shared images is unavailable. Therefore the current study presents a secret image-sharing scheme to efficiently detect cheaters in secret keys and shared images. A significant aspect of the proposed scheme relies on not needing a one-way hash function because a security issues existed in hash functions.[17]

The rest of this paper is organized as follows. Section 2 reviews important secret image-sharing schemes on detecting cheaters. Section 3 introduces the proposed tamper-proof secret image-sharing scheme. Algorithms of initial procedure, sharing a secret image to shared images, recovering with verification from secret keys and shared images, and security analysis are presented in Secs. 3.1–3.4, respectively. Section 4 provides experimental results and comparisons between the proposed scheme and other methods. Section 5 offers a conclusion with suggestions for future research.

## 2 Literature Review

This section reviews the literatures on cheater detection of secret image-sharing problems. We review the publications of Chen and Suen[18] and Zhao et al.[19] in Secs. 2.1 and 2.2, respectively.

### 2.1 *Review of the Chen and Suen's Secret Image-Sharing Scheme*

Chen and Suen[18] adopted the Wu and Wu[15] plan—which is based on a one-way hash function h, a selected prime number $P$, and a calculated large number $T$—to identify cheaters in a

Paper 12274 received Jul. 19, 2012; revised manuscript received Nov. 15, 2012; accepted for publication Dec. 11, 2012; published online Jan. 9, 2013.

secret sharing scheme. Both sharing and recovering strategies are examined in the study of Chen and Suen $(t, n)$ scheme.

In the sharing procedure, the secret image is shared with n shared images $y_1, y_2, \ldots, y_n$ using the Shamir secret sharing method. Then a large number $T$ is calculated and publicly accessed, where $T = \sum_{i=1}^{n} h(y_i)p^{2(i-1)} + \sum_{i=1}^{n-1} cp^{2i-1}$, $1 \leq c < p$ and $h()$ is a hash function.

During recovering procedure, all collected shared images $y_j^*$ are checked by functions $T_j^* = \sum h(y_j^*)p^{2(j-1)}$ and $\left\lfloor T - T_j^*/p^{2(j-1)} \right\rfloor (\mathrm{mod} p) = 0$ to determine whether $y_j^*$ is a cheating shared image. Then the Lagrange interpolation method is applied to recover the secret image $s$, when the number of correct $y_j^*$ is $t$.

### 2.2 *Review of the Zhao et al.'s Secret Image-Sharing Scheme*

Zhao et al.[19] applied Thien and Lin's[1] secret image-sharing scheme for sharing a secret image and verified it by modulus calculation. Assume that $H$ is the secret image keeper and $P_i$ $(i = 1, \ldots, n)$ denotes each participant. Three procedures—initial, sharing, and recovering—are needed in their approach.

During the initial process, the keeper $H$ publishes two parameters $\{g, n_0\}$. Then each participant $P_i$ selects his or her secret key $s_i$ and calculates his or her public parameter $r_i$.

When sharing the secret image, the keeper $H$ calculates two parameters $r_0$ and $w_i$, and then chooses two other parameters, $\{r_0, f\}$, for sharing the image. $H$ then calculates the shared message by $h_j(w_i) = (b_0 + b_1 w_i + \ldots + b_{t-1} w_i^{t-1}) \bmod 251$, where $b_0, b_1, \ldots, b_{t-1}$ are pixel values. Then $H$ publishes $\{h_j(w_t)\}$.

During the recovering with verification procedure, each participant $P_i$ calculates the checked message $w_i'$. If $w_i = w_i'$, $H$ confirms participant $P_i$ by providing a verified secret key, and the secret image can be reconstructed. Without verification, $P_i$ is a confirmed cheater. The correct reconstructed image is then calculated using the Lagrange interpolation method.

In Zhao et al.'s scheme, the accuracy of a shared image relies on its corresponding secret key, rather than checking content of the shared image itself. This creates a gap in the security. Therefore we present a secure secret image-sharing scheme that checks the validity of secret key and the shared image.

## 3 Proposed Scheme

This section introduces the proposed $(t, n)$ tamper-proof secret image-sharing scheme. Assume that $P_i$ $i = 1, 2, \ldots, n$, denotes each participant. In Sec. 3.1, the proposed scheme first allows each participant to configure his secret key. Section 3.2 presents a description of the image sharing process. Section 3.3 shows the verification and reconstruction processes for keys and shared images. Section 3.4 analyzes the security of the proposed scheme.

### 3.1 *Initial Algorithm*

This section uses exponent and modulus computation to determine each participant's secret key. Steps of initial algorithm are illustrated as follows.

1. The dealer selects two prime numbers, $p_0$ and $q_0$, and calculates $n_0 = p_0 \times q_0$.
2. The dealer selects an integer $g_0$, satisfying gcd $(g_0, n_0) = 1$ and then publishes $\{n_0, g_0\}$.
3. Each participant $P_i$ chooses two prime numbers $p_i$ and $q_i$, and then calculates their product $n_i$, denoted by $n_i = p_i \times q_i$. $P_i$ chooses another integer $g_i$, satisfying gcd $(g_i, n_i) = 1$, and then calculates its multiplicative inverse $f_i$, satisfying $g_i \times f_i = 1 \bmod (p_i - 1) \times (q_i - 1)$.
4. $P_i$ publishes $\{g_i, n_i\}$.
5. Participant $P_i$ takes $p_i$ as his or her secret key and sends $f_i$ and $r_i$ to the dealer, where $r_i = g_0^{p_i} \bmod n_0$.

The dealer should preserve each received $r_i$ differently, which means that each $P_i$ possesses different secret key $p_i$, to distinguish participant's role. This extra step requires participants possessing identical $r_i$, to repeat steps 3 to 5 to obtain new secret key. Furthermore, the prime number $p_i$ is the secret key that $P_i$ possesses, whereas the dealer retains $r_i$ instead of $p_i$.

### 3.2 *Sharing Algorithm*

When sharing the image, the dealer should first calculate new key $w_i$ for each participant. The following algorithm illustrates the steps taken during the sharing process.

1. The dealer randomly selects an integer $s_0 \in [2, n_0]$, satisfying gcd $[s_0, (p_0 - 1)] = 1$ and gcd $[s_0, (p_0 - 1)] = 1$.
2. The dealer computes $r_0$ and $w_i$, where $r_0 = {g_0}^{s_0} \bmod n_0$, $w_i = {r_i}^{s_0} \bmod n_0$.
3. The dealer sends $r_0$ to each participant $P_i$.
4. The dealer partitions the secret image to blocks of $t$ pixels, where $B_k$ $k = 1, \ldots, r$ denotes each partitioned block and $r$ is the block number. The dealer then applies to each block $B_k$ the following steps.

   4.1. Replace $B_k$ by $B_k \bigoplus R_k$, where $R_k$ is a random block and $\bigoplus$ denotes Exclusive-OR operation.

   4.2. It constructs a polynomial function $f_k(x) = (b_0 + b_1 x + \ldots + b_{t-1} x^{t-1}) \bmod 251$, where $b_0, b_1, \ldots, b_{t-1}$ represent $t$ pixels in one $B_k$ block.

   4.3. The dealer calculates $y_{i,k} = f_k(w_i)$, where $w_i$ $(i = 1, 2, \ldots, n)$ is obtained from step 2.

   4.4. The dealer calculates $x_{i,k} = y_{i,k}^{g_i} \bmod n_i$ for the shared image belonging to participant $P_i$.

   4.5 The dealer randomly selects a prime number $c$ and computes $h_{i,k} = cy_{i,k} \bmod n$, where $n$ is a number defined by larger than number of participants and mod $(c, n) = 1$.

   4.6. The dealer randomly selects a prime number $a$ and computes $T_k = \sum_{i=1}^{n} h_{i,k} a^i$, with $a > n$.

5. The dealer sends shared image $X_i$, which is formed from $x_{i,k}$ $(k = 1, 2, \ldots, r)$, to participants $P_i$ and publishes $\{T_k\}$.

Since in conventional Shamir-Lagrange method, a prime number is needed and the number is determined by 251 in the proposed scheme. Therefore all parameters $b_i$ in step 4.2 must be restricted between 0 and 250. However, largest pixel value is 255. Consequently, this gap can be solved by Thien and Lin's method.[1] For an image pixel $g$, $g$ will be partitioned to two numbers 250 and $g$-250 if $250 \leq g \leq 255$. Two numbers 250 and $g$-250 represent two parameters $b_i$ in step 4.2.

### 3.3 *Recovering with Verification Algorithm*

This section presents the verification algorithm that accompanies the image recovery process. First the dealer verifies the authenticity of each participant's possessing key $p_i$ and shared image $X_i$. Then the dealer uses these secret keys and shared images to reconstruct the secret image. The following is an algorithm for recovering with verification.

1. The dealer acquires the participant's shared image $X_i$ to calculate the original shared message $y_{i,j}$ by $y_{i,j} = x_{i,j}^{f_i} \bmod n_i$, where $x_{i,j}$ is the $j$'th number in $X_i$.
2. The dealer employs the participant $P_i$'s secret key $p_i$ and $y_{i,j}$ to verify $P_i$'s authenticity by checking whether $w_i$ is equal to $w'_i$ ($w'_i = r_0^{p_i} \bmod n_0$) and whether $h_{i,j}$ ($h_{i,j} = cy_{i,j} \bmod n$) is equal to $h'_{i,k}$ ($h'_{i,k} = \lfloor T_k / a^i \rfloor \bmod a$).
3. When all participants are authenticated, the following Lagrange interpolation method on each set of secret keys and shared messages $(w_i, y_{i,j})$ is calculated by

$$f_k(x) = \sum_{i=1}^{t} y_{i,j} \prod_{j=1}^{t} \frac{x - w_j}{w_i - w_j} \bmod 251$$
$$= (b_0 + b_1 x + \cdots + b_{t-1} x^{t-1}) \bmod 251,$$

   where coefficients $b_0, b_1, \ldots, b_{t-1}$ represent pixels of one secret image block $B_k$.
4. Replace $B_k$ by $B_k \bigoplus R_k$, where $R_k$ is the random block used in sharing algorithm.
5. Combine all $B_k$ blocks to acquire the reconstructed secret image.

Note that step 3 is performed when all keys $p_i$ and shared images $X_i$ are verified.

### 3.4 *Security Analysis*

This section analyzes the security of the proposed tamper-proof secret image-sharing scheme. First we will check whether any cheated modification on secret key or shared image can be well detected. Then, since the proposed scheme adopts exponent and modulus computation, we also analyze the common modulus attack in this section.

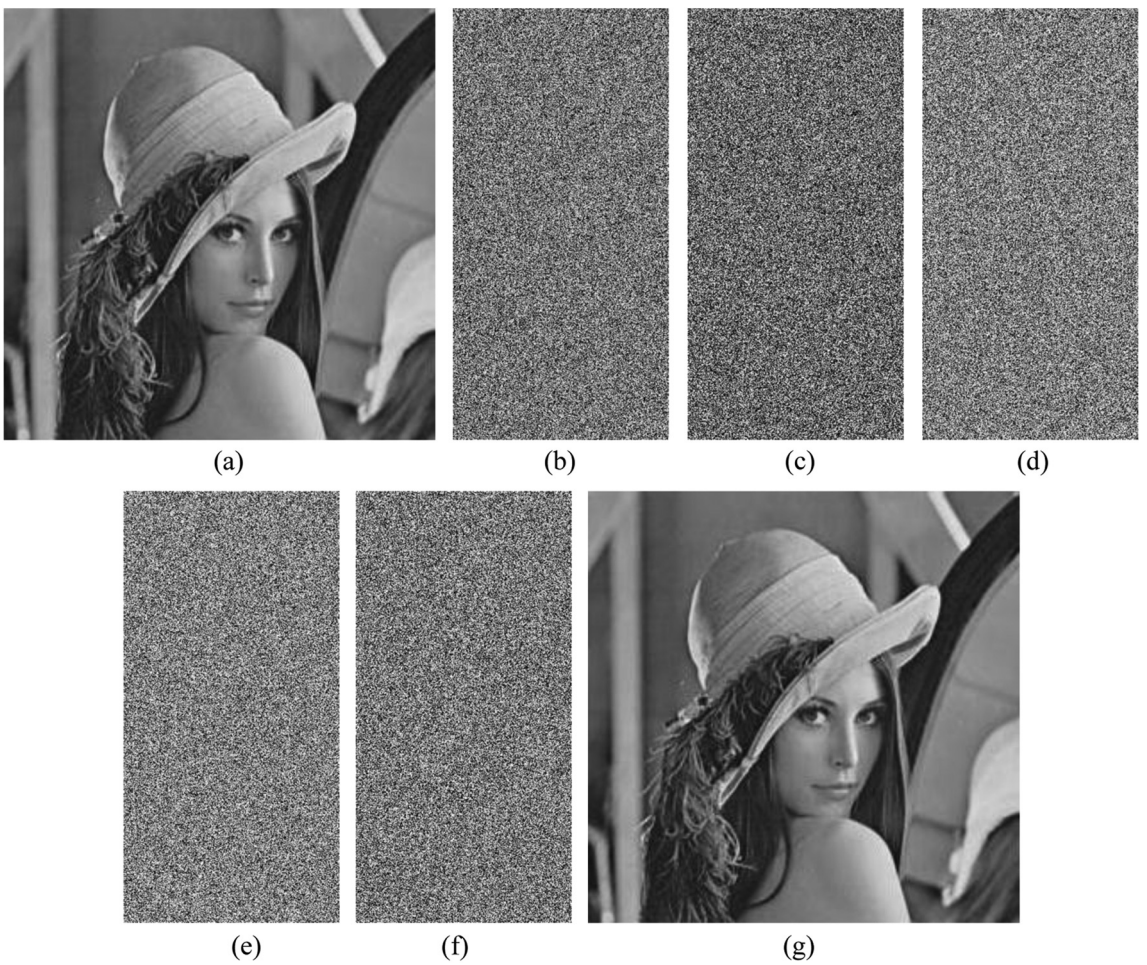
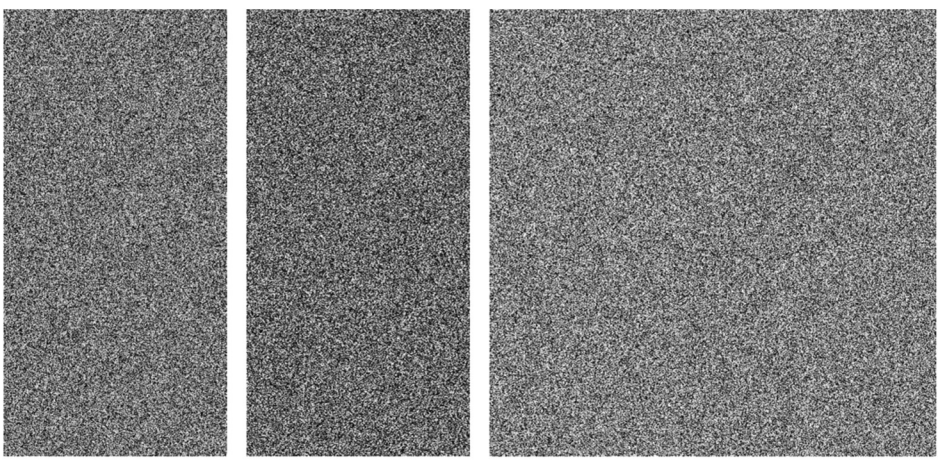



**Fig. 1** (a) secret image; (b) to (f) five shared images; (g) reconstructed image from (b) and (c).

**Table 1** The parameters used in the experiments.

| Procedure | Parameters | Values | Public parameters | Values |
|---|---|---|---|---|
| Initial algorithm | $(p_0, q_0)$ | (5,7) | $n_0$ | 35 |
| | $g_0$ | 17 | $g_0$ | 17 |
| | $(p_1, q_1, r_1)$ | (11, 43, 33) | $(g_1, n_1)$ | (7, 473) |
| | $(p_2, q_2, r_2)$ | (17, 19, 12) | $(g_2, n_2)$ | (11, 323) |
| | $(p_3, q_3, r_3)$ | (31, 37, 3) | $(g_3, n_3)$ | (13, 1147) |
| | $(p_4, q_4, r_4)$ | (37, 41, 17) | $(g_4, n_4)$ | (17,1517) |
| | $(p_5, q_5, r_5)$ | (57, 23, 27) | $(g_5, n_5)$ | (19, 1311) |
| Sharing algorithm | $s_0$ | 7 | | |
| | $r_0$ | 3 | | |
| | $w_1$ | 12 | | |
| | $w_2$ | 33 | | |
| | $w_3$ | 17 | | |
| | $w_4$ | 3 | | |
| | $w_5$ | 13 | | |
| | $A$ | 47 | | |

The secret key is verified by exponential computation. In step 2 of the recovering with verification algorithm, $w_i' = r_0^{p_i}$, where $r_0 = g_0^{s_0}$ is defined in step 2 of the sharing algorithm. Therefore $w'_i = r_0^{p_i} = (g_0^{s_0})^{p_i} = (g_0^{p_i})^{s_0} = r_i^{s_0} = w_i$, since $r_i = g_0^{p_i}$, as defined in step 5 of the initial algorithm. Consequently, the accuracy of the proposed verification procedure is proved. For any cheated secret key as defined by replacing $p_i$ by $p_i'$ satisfying $p_i' \neq p_i$, the verification becomes checking whether cheated $w_i' = r_0^{p_i'} = (g_0^{s_0})^{p_i'} = (g_0^{p_i'})^{s_0}$ and $w_i = r_i^{s_0} = (g_0^{p_i})^{s_0}$ are the same. Note that all these computations are calculated under mod $n_0$. This equivalence verification can be described as checking whether $(g_0^{p_i'} \bmod n_0)$ is equivalent to $(g_0^{p_i} \bmod n_0)$. Thus a participant $P_i$ can choose another secret key $p_i'$ satisfying $g_0^{p_i'} = g_0^{p_i} \bmod n_0$. However, when $n_0$ is a very large number, $p_i'$ is



(a) (b) (c)

**Fig. 2** (a) shared image in Fig. 1(b); (b) shared image in Fig. 1(c); (c) reconstructed image from (a) and (b) with cheated $w_1 = 23$ and $w_2 = 33$.

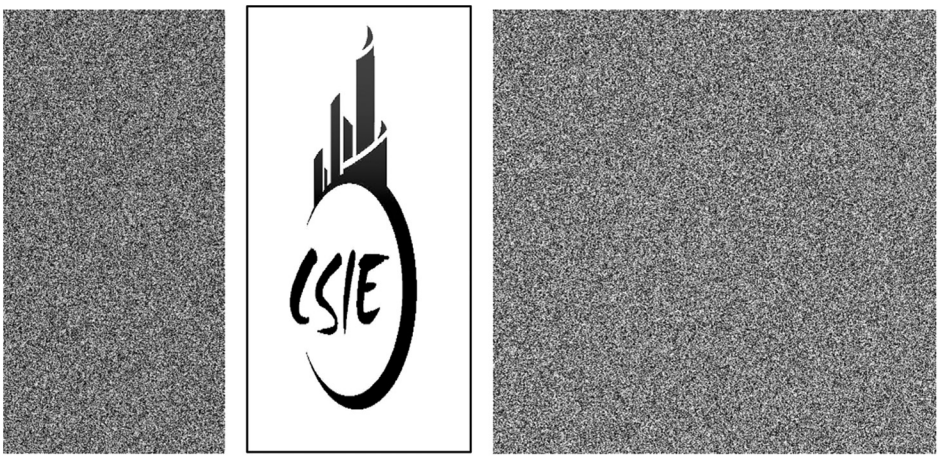



**Fig. 3** (a) shared image in Fig. 1(b); (b) cheated shared image in Fig. 1(c); (c) reconstructed image from (a) and (b) with $w_1 = 12$ and $w_2 = 33$.

hard to be found.[20] Moreover, an attacker can only cheat $r_i$ and acquire $g_0$ and $n_0$ over networks. He has to find $p_i$ from $r_i = g_0^{p_i} \bmod n_0$ and it's also a hard work when $n_0$ is a very large number. Since a $p'_i$ satisfying the equivalence verification is hard to be found, any cheated secret key will always be detected.

The shared image is verified by public data $T_k$ and exponent computation. For an attacker, he cannot find $n$ in step 4.4 of the sharing algorithm. Therefore the calculated $h_{i,j}$ from his cheated shared message $x'_{i,j}$ and the following computations, $h_{i,j} = cy_{i,j} \bmod n$ and $y_{i,j} = x'^{f_i}_{i,j} \bmod n_i$, is very hard to be equal to the original $h_{i,j}$ obtaining from public data $T_k$. Consequently, the shared image is hard to be replaced by any cheated shared image.

The common modulus attack[20] indicates that secret message $m$ can be recovered by two secret keys $e_1$ and $e_2$ corresponding with two shared messages $m_1 = m^{e_1}$ and $m_2 = m^{e_2}$, respectively. Since $e_1$ and $e_2$ are relatively prime, there are two numbers $a_1$, $a_2$ such that $a_1e_1 + a_2e_2 = 1$. Therefore the computation is then obtained as following equations $(m_1)^{a_1} \cdot (m_2)^{a_2} = (m^{e_1})^{a_1} \cdot (m^{e_2})^{a_2} = m^{a_1e_1+a_2e_2} = m$. Note that all above computations are calculated under mod $N$.

In the proposed scheme, the shared messages for participants $P_1$ and $P_2$ are $[f_k(r_1^{s_0})]^{g_1} \bmod n_1$ and $[f_k(r_2^{s_0})]^{g_2} \bmod n_2$, respectively. Since $f_k(r_1^{s_0}) \neq f_k(r_2^{s_0})$, and $n_1 \neq n_2$, the common modulus attack cannot be mounted by anyone who has only two secret messages. This property also shows that $[f_k(r_1^{s_0})]^{a_1g_1} \cdot [f_k(r_2^{s_0})]^{a_2g_2}$ cannot recover the coefficients $b_i$ in $f_k(x) = (b_0 + b_1x + \ldots + b_{t-1}x^{t-1})$, even though $a_1g_1 + a_2g_2 = 1$. Furthermore, since $t$ shared messages corresponding with secret keys meets the proposed $(t, n)$ thresholds, so we can obtain these $b_i$ coefficients.

## 4 Experimental Results and Discussion

### 4.1 *Experimental Results*

This section presents the experimental results obtained from the proposed method. The test image is LENA with a size of $512 \times 512$, and the selected thresholds are (2, 5). This threshold assignment shares the secret image with five participants, and collecting any two correct participants' secret keys with shared images recovers the secret image. Figure 1(a) shows the secret image LENA with a size of $512 \times 512$ and Fig. 1(b) to 1(f) shows five shared images corresponding with secret keys, as defined in Table 1. The set thresholds of (2, 5) acquire a shared image with size $512 \times 256$.

Figure 2 uses a cheated secret key $w_1 = 23$, instead of correct $w_1 = 12$, to recover the secret image. Since the secret key is wrong, the cheated secret key will be detected in the proposed scheme. If we ignore the wrong detection in step 2 of recovering with verification algorithm, we acquire the recovered secret image as shown in Fig. 2(c). Another experiment on cheated shared image is illustrated in Fig. 3. Figure 3(b) shows the cheated shared image, where Fig. 3(a) and all secret keys are correct. Ignoring the cheated shared image detection and keeping calculation acquire the reconstructed secret image as shown in Fig. 3(c). In these two figures, we find that any cheated secret key or shared image

**Table 2** Characteristics comparison between the proposed scheme and important literatures.

| Features | |
|---|---|
| **Schemes** | **Features of scheme** |
| Ref. 2 | Multisecret images sharing |
| Ref. 18 | Shared image verification |
| Ref. 3 | Progressive reconstruction |
| Ref. 8 | Visual cryptography and secret image sharing |
| Ref. 1 | Secret image sharing |
| Ref. 21 | Scalable shared image |
| Ref. 11 | Shared image size constraint |
| Ref. 19 | Secret key verification |
| The proposed scheme | Secret key and shared image verification |

**Table 3** Comparisons with other secure secret image-sharing schemes.

| Features / Schemes | No hash function | Secret key verification | Shared image verification | Network usage | Dealer possessing load | Public sharing load |
|---|---|---|---|---|---|---|
| Ref. 18 | × | × | ✓ | × | × | A large number $T$ |
| Ref. 19 | ✓ | ✓ | × | × | $N+2$ coefficients | × |
| The proposed scheme | ✓ | ✓ | ✓ | ✓ | $N+4$ coefficients | numbers $\{T_k\}$, $2n+3$ coefficients |

causes the wrong reconstructed secret image. Note that the proposed scheme can detect any cheated secret key or shared image efficiently. Therefore the wrong reconstructed secret image such as Figs. 2 or 3 will not be acquired in the proposed scheme.

### 4.2 *Comparisons and Discussion*

The proposed scheme verifies participants in secret image sharing problem. Two comparisons are provided in this section. First an overall comparison between the proposed scheme and other important works[1–3,8,11,18,19,21] is listed in Table 2. Second, a comparison of the secret image-sharing schemes with cheater identification properties is shown in Table 3.

Table 2 shows a comparison of characteristics between these propositions. These characteristics include sharing multiple images,[2] image verification,[18] progressive,[3] visual cryptography and secret image sharing,[8] perfect secret image sharing,[1] scalability,[21] size constraints,[11] secret key verification,[19] and secret key and shared-image verification proposed in this paper.

Table 3 shows a comparison of the results between the proposed scheme and other secure secret image-sharing schemes.[18,19] Four conclusions are drawn from this table. First, the proposed scheme verifies both secret keys and shared images, which perform better than previous studies[18,19] that verify only either the shared image or secret key. Second, the required parameters loads, including dealer processing and public sharing, are few more than required by Refs. 18 and 19. Third, the extra load is cause by the free hash function, and the extra load is limited. At last, secret key selection is determined by participant and this process can be done over networks. Therefore, Tables 2 and 3 show that the proposed scheme has significant property of detecting cheaters both in secret key and shared image.

## 5 Conclusions

This paper presents a secret image-sharing scheme with the properties of detecting cheaters both in secret key and shared image. The proposed scheme presents three algorithms: initial, sharing, and recovering with verification. The strategy for key validation is different from previous works. We allow each participant to select his or her secret key, and the dealer checks the validity of each key. Verification during image recovery is also based on the participant's selected secret key. This property of determining secret key from a participant fits the network requirement well. Security analysis and experimental results demonstrate that the proposed scheme behaves strong security coverage. Future work will focus on combining other characteristics such as multiple image sharing to enhance the benefits of the proposed scheme.

### *Acknowledgments*

The authors gratefully acknowledge the helpful comments and suggestions of the reviewers. This work was supported in part by the National Science Council project under Grant NSC 100-2221-E-032-056.

### *References*

1. C. C. Thien and J. C. Lin, "Secret image sharing," *Comput. Graph.* **26**(5), 765–770 (2002).
2. C. C. Chen and Y. W. Chien, "Sharing numerous images secretly with reduced possessing load," *Fundamenta Inform.* **86**(4), 447–458 (2008).
3. S. K. Chen and J. C. Lin, "Fault-tolerance and progressive transmission of images," *Pattern Recognit.* **38**(12), 2466–2471 (2005).
4. W. P. Fang, "Friendly progressive visual secret sharing," *Pattern Recognit.* **41**(4), 1410–1414 (2008).
5. C. P. Huang, C. H. Hsieh, and P. S. Huang, "Progressive sharing for a secret image," *J. Syst. Software* **83**(3), 517–527 (2010).
6. K. H. Hung, Y. J. Chang, and J. C. Lin, "Progressive sharing of an image," *Opt. Eng.* **47**(4), 047006 (2008).
7. S. J. Lin, L. S. Chen, and J. C. Lin, "Fast-weighted secret image sharing," *Opt. Eng.* **48**(7), 077008 (2009).
8. S. J. Lin and J. C. Lin, "VCPSS: A two-in-one two-decoding-options image sharing method combining visual cryptography (VC) and polynomial-style sharing (PSS) approaches," *Pattern Recognit.* **40**(12), 3652–3666 (2007).
9. C. N. Yang and C. B. Ciou, "Image secret sharing method with two-decoding-options: lossless recovery and previewing capability," *Image Vis. Comput.* **28**(12), 1600–1610 (2010).
10. C. C. Thien and J. C. Lin, "An image-sharing method with user-friendly shadow images," *IEEE Trans. Circ. Syst. Video Technol.* **13**(12), 1161–1169 (2003).
11. Y. S. Wu, C. C. Thien, and J. C. Lin, "Sharing and hiding secret images with size constraint," *Pattern Recognit.* **37**(7), 1377–1385 (2004).
12. C. C. Chen and W. Y. Fu, "A geometry-based secret image sharing approach," *J. Inform. Sci. Eng.* **24**(5), 1567–1577 (2008).
13. T. H. Chen and C. S. Wu, "Efficient multi-secret image sharing based on Boolean operations," *Signal Process.* **91**(1), 90–97 (2011).
14. S. J. Shyu and Y. R. Chen, "Threshold secret image sharing by Chinese remainder theorem," in *Proc. IEEE Asia-Pacific Services Computing Conf.*, pp. 1332–1337, IEEE, Yilan, Taiwan (2008).
15. T.-C. Wu and T.-S. Wu, "Cheating detection and cheater identification in secret sharing schemes," *IEE Proc. Comput. Dig. Tech.* **142**(5), 367–369 (1995).
16. C.-C. Chang and R.-J. Hwang, "Efficient cheater identification method for threshold schemes," *IEE Proc. Comput. Dig. Tech.*, **144**(1), 23–27 (1997).
17. K. J. Tan, H. W. Zhu, and S. J. Gu, "Cheater identification in ($t$, $n$) threshold scheme," *Comput. Commun.* **22**(8), 762–765 (1999).
18. C. C. Chen and G. Y. Suen, "Sharing an image with cheater identification," *Int. J. Innovat. Comput. Inform. Control* **6**(2), 677–685 (2010).
19. R. Zhao et al., "A new image secret sharing scheme to identify cheaters," *Comput. Stand. Interfac.* **31**(1), 252–257 (2009).
20. M. J. Hinek, *Cryptanalysis of RSA and Its Variants*, Chapman and Hall/CRC, Taylor & Francis Group, Boca Raton, Florida (2010).
21. R. Z. Wang, Y. F. Chien, and Y. Y. Lin, "Scalable user-friendly image sharing," *J. Vis. Comm. Image Represent.* **21**(7), 751–761 (2010).

**Chien-Chang Chen** received a BS from the Department of Computer and Information Science at Tunghai University, Taiwan, in 1991, and a PhD from the Department of Computer Science at National Tsing Hua University, Taiwan, in 1999. He is currently an associate professor at the Department of Computer Science and Information Engineering, Tamkang University, Taiwan. His research interests include secret image sharing, watermarking, and texture analysis.

**Chong-An Liu** received an MS from the Department of Computer Science and Information Engineering, Tamkang University, in 2012. His research interests include secret image sharing.