A High Quality Image Sharing with Steganography and Adaptive Authentication Scheme

Chia-Chun Wu [a], Shang-Juh Kao [a] and Min-Shiang Hwang [b,*]

[a] Department of Computer Science and Engineering,
National Chung Hsing University
No. 250, Kuo Kuang Road, Taichung 402, Taiwan
phd9420@cs.nchu.edu.tw and sjkao@cs.nchu.edu.tw

[b] Department of Computer Science & Information Engineering,
Asia University
No. 500, Lioufeng Road, Wufeng, Taichung 413, Taiwan
mshwang@asia.edu.tw ([*]Corresponding author)

Abstract

With the rapid growth of numerous multimedia applications and communications through Internet, secret image sharing has been becoming a key technology for digital images in secured storage and confidential transmission. However, the stego-images are obtained by directly replacing the least-significant-bit planes (LSB) of cover-images with secret data and authentication code in most schemes, which will result in the distortion of the stego-images. In this paper, we proposed a novel secret image sharing scheme by applying optimal pixel adjustment process to enhance the image quality under different payload capacity and various authentication bits conditions. The experimental results showed that the proposed scheme has improved the image quality of stego images by 4.71%, 9.29%, and 11.10%, as compared with the schemes recently proposed by Yang *et al*., Chang *et al*., and Lin and Tsai. We also provide several experiments to demonstrate the efficacy of authentication capability of the proposed scheme. In other words, our scheme maintains the secret image sharing and authentication ability while enhances the image quality.

Keywords*: Steganography, Cryptography, Secret Sharing, Authentication Code, Fragile Watermark,* (*k, n*)*-Threshold Scheme, Optimal Pixel Adjustment, Hash Message Authentication Code*

1. Introduction

Along with the quick development of communication and network, copying, publishing and transmitting digital multimedia via the Internet is quite convenient. How effective processing and management of sensitive information have become an important topic to be considered nowadays. Most of the information security issues could be used by cryptography and steganography technologies. In cryptography, security is dependent on the length of the secret keys. Normally, cryptography is used in digital communications, computer networks, and computer security. Steganography is usually applied in various digital media types such as image, video, and audio nowadays.

In 1979, Shamir proposed the theory of secret sharing scheme based on Lagrange's polynomial interpolation [1, 2]. It allows the sharing of a secret message among a group of participants. The basic idea is to split a secret message *s* into *n* shares, such that for any *k* shares the message *s* could be determined, where *k* is used as a threshold and $k \leqq n$. The scheme and theory can be defined as follows:

(1) Secret *s* is to be divided into *n* shares, and a part of secret message is called a share.

(2) Given any *k* or more out of *n* shares, a given secret *s* would be easy to reconstruct.

(3) If an illegal user only has the knowledge of any *k*–1 or fewer shares, then no information about the secret *s* could be determined.

A mechanism of secret sharing scheme is desirable for situations where permission to access the secret or information message depends not on an individual but on a group of people. The main advantage of the secret sharing concept gives a good solution for data security because all participants are required to break the secret data into several pieces and keep a secret share independently. Security can be achieved through the ownership of the secret share held together by participants. Typically, such a method is called a (*k*, *n*)-threshold secret sharing scheme.

There are many real-life applications, for example, it might be necessary in a company that the managers share the digital documents, and only when any two or more out of all managers work together with mutual agreement can they see the digital documents. Another application is like the case in a bank that a password was broken into three pieces, and only by more than one teller can they open a vault. The secret share of digital documents or passwords can be transformed and shared by image. In order to prevent honest participants from recovering the disordered message or providing a fake image by a cheater, the authentication ability is required in such applications. This concept of secret sharing gives a good solution to the requirements of both security protection and identity authentication.

In 2002, Thien and Lin [3] proposed a (*k*, *n*)-threshold secret image sharing scheme that produces smaller noise-like shadow images. Secret image can be shared by several shadow images so the size of each shadow image is only 1/*k* of that of the secret image for convenient transmission, storage, and hiding in their scheme. In order to identify and manage the shadow images with convenience, they also suggested another user-friendly image sharing method such that the shadow images look like natural image in 2003 [4].

In 2004, Lin and Tsai [5] proposed a novel secret image sharing method that is based on the Shamir's (*k*, *n*)-threshold scheme. By using the parity check bit, they claimed that their scheme can prevent from incidentally bringing an erroneous stego-image or intentionally providing a false image to achieve the authentication goal. Afterward Yang *et al*. proposed an improved scheme to overcome the three weaknesses: image authentication by dishonest participant, deterioration quality of stego-image, and non-lossless secret image scheme for secret image in Ref. [6]. Recently, Chang *et al*. [7] also proposed another scheme to improve the authentication ability and visual image quality.

The simple least-significant-bit substitution (LSBs) [8, 9] method is to produce high distortion (or error) in those schemes, and there is a common problem of those schemes for embedding secret data and authentication code in cover images. [10] Therefore, we intended to design a novel secret image sharing scheme to improve the quality of stego-images, which employs Chan and Cheng's simple LSBs substitution with an optimal pixel adjustment process

[9, 11-13] in this paper. The experimental results showed that the proposed scheme has provided significantly better image quality than the others. The capability of identifying the tampered region under various authentication bits conditions is also estimated to demonstrate that our scheme has high authentication ability.

Unlike all previous methods, we use a distinct image identification number as the input of polynomial and keep it as a private key for each participant. Only the legal participants know the image identification number. Actually, it is difficult to retrieve an image identification number from stego-images for an illegal user because the private keys are not present.

The remainder of this paper is organized as follows. In Section 2, we describe the Yang *et al*. and Chang *et al*. secret image sharing schemes. In Section 3, we will introduce our novel secret image sharing scheme, then an example that demonstrates how the optimal LSBs method decreases the distortion caused by the simple LSBs method will also described. We have analyzed the visual image quality and the authentication capability, and the experimental results are given in Section 4. Finally, we give a discussion and briefly conclude this paper in Section 5.

2. Related work

Two secret image sharing techniques would be briefly introduced in this section. The former secret image sharing scheme proposed by Yang *et al*. is described in Section 2.1. The latter improved scheme proposed by Chang *et al*. is described in Section 2.2.

Before introducing the related work, some assumptions are supposed as follows. Assume that a grayscale secret image S of size $m \times m$ is to be protected by embedding secret share messages and fragile watermark [14-22] signal bits into n ordinary cover images. The secret image S is to be divided into $m \times m$ sections. They further assumed that the i^{th} secret pixel s_i is a single integer value from $m \times m$ secret image $S = \{s_1, s_2, \ldots, s_{m\times m}\}$, which is shared by n participants. They supposed there are n user-selected ordinary cover images $I^{(j)}=\{I^{(1)}, I^{(2)}, \ldots, I^{(n)}\}$ for a group of n participants $P^{(j)}=\{P^{(1)}, P^{(2)}, \ldots, P^{(n)}\}$, of which the size is $2m \times 2m$. In general, the size of secret image is 256 × 256 pixels and the size of cover image is 512 × 512 pixels. Each of them is divided into nonoverlapping 2 × 2 blocks (denoted as $B_i^{(j)}$), where $1 \leqq i \leqq m \times m$ and $1 \leqq j \leqq n$. The four pixels in each block $B_i^{(1)}$, $B_i^{(2)}$, …, $B_i^{(n)}$ are to be denoted as X_i, W_i, V_i, and U_i. Also, x_i, w_i, v_i, and u_i represent their binary values respectively. An illustration of the locations of the pixels value in each block $B_i^{(j)}$ is shown in Figure 1. Each stego-block $B_i^{"(j)}$ is then found from secret shares and signal bits. Finally, n stego-images $I^{*(j)} = \{I^{*(1)}, I^{*(2)}, \ldots, I^{*(n)}\}$ with $2m \times 2m$ size are obtained by n participants until all pixels of the secret image are processed.

X_i $x_i=(x_{i8}x_{i7}x_{i6}x_{i5}x_{i4}x_{i3}x_{i2}x_{i1})_2$	W_i $w_i=(w_{i8}w_{i7}w_{i6}w_{i5}w_{i4}w_{i3}w_{i2}w_{i1})_2$
V_i $v_i=(v_{i8}v_{i7}v_{i6}v_{i5}v_{i4}v_{i3}v_{i2}v_{i1})_2$	U_i $u_i=(u_{i8}u_{i7}u_{i6}u_{i5}u_{i4}u_{i3}u_{i2}u_{i1})_2$

Figure 1. The representation of the 4 pixels in each 2 × 2 block $B_i^{(j)}$

2.1. Review of Yang *et al.*'s scheme

Yang *et al*. [6] rearranged the integer values $x_i^{(j)}$, secret shares $F_i^{(j)}$, and hash bits $p_i^{(j)}$ to improve the stego-images quality. The secret shares $F_i^{(j)}$ and hash bits $p_i^{(j)}$ are divided and embedded equally in each block $B_i^{(j)}$. Furthermore, the eight bits $(x_{i8}x_{i7}x_{i6}x_{i5}x_{i4}x_{i3})_2$ and $(v_{i4}v_{i3})_2$ in each block $B_i^{(j)}$ are combined to form an integer values $x_i^{(j)}$ as the input of polynomial. The integer values $x_i^{(j)}$ can be computed by the following equation:

$$\begin{aligned} x_i &= (x_{i8}x_{i7}x_{i6}x_{i5}x_{i4}x_{i3}v_{i4}v_{i3})_2 \\ &= [(x_{i8}x_{i7}x_{i6}x_{i5}x_{i4}x_{i3}x_{i2}x_{i1})_2 \; AND \; 11111100_{(2)}] \\ &+ [(v_{i8}v_{i7}v_{i6}v_{i5}v_{i4}v_{i3}v_{i2}v_{i1})_2 \; AND \; 00001100_{(2)}] \; / \; 2^2, \end{aligned} \qquad (1)$$

where "*AND*" operator is referred to as the bitwise binary operation. For all n integer values, $x_i^{(1)}$, $x_i^{(2)}$, …, $x_i^{(n)}$, $x_i^{(j)}$ must be distinct from the others and $x_i^{(j)} \neq 0$. If the input of polynomial $x_i^{(j)}$ are repeated, for example, it can modify bit v_{i3} of integer value $v_i^{(j)}$ to form $v_i'^{(j)}$ to satisfy the condition. Hence, the new integer values $x_i'^{(j)}$ are obtained.

The secret pixel values greater than 250 are modified to 250 in Lin and Tsai's scheme. In order to construct a lossless version of image sharing scheme for secret image, Yang *et al*. used the power-of-two Galois Field GF (2^8) [23, 24] instead of prime Galois Field GF (251) [6, 25]. Hence, the improved (k–1)-degree polynomial can be defined as follows:

$$F(x) = (s + a_1 \times x^1 + \ldots + a_{k-1} \times x^{k-1}) \bmod \mathrm{GF}(2^8), \qquad (2)$$

where x is a pixel value chosen from the cover images, s is a single integer secret pixel chosen from the secret image, a_1, a_2, …, a_{k-1} are random numbers, and "mod" operator is referred to as the modulo operation. All of these value of x, s, and a_1, a_2, …, a_{k-1} are between 0 and 255. Any secret pixels s_i larger than 250 does not distort in secret image. For each unique value $x_i'^{(j)}$, the integer value of $F(x_i'^{(j)})$ by Eq. (2) is computed to form a secret share $F_i^{(1)}$, $F_i^{(2)}$, …, $F_i^{(n)}$, respectively. Each pair of $(x_i'^{(j)}, F(x_i'^{(j)}))$ is a secret share of secret s_i.

Yang *et al*. also confirmed that dishonest participants' problem in Lin and Tsai's scheme. They attempted to improve this situation by using the hash function with secret key K_1, $H_{K1}(\cdot)$, block index B_{id}, and stego-image identification $I_{id}^{(j)}$ to compute the n hash bits $p_i^{(1)}$, $p_i^{(2)}$, …, $p_i^{(n)}$ for each block $B_i^{(j)}$. After that, n secret shares $F_i^{(j)}$ and n hash bits $p_i^{(j)}$ are embedded into four pixels X_i, W_i, V_i, and U_i of each block $B_i^{(j)}$ by simple LSBs embedding method to form $B_i''^{(j)}$, respectively. Figure 2 shows the results of applying Yang *et al*.'s secret image sharing scheme with a single image pixel s_i as the secret to block $B_i^{(j)}$, where F_{i8}, F_{i7}, …, F_{i1} are binary formats of the secret share $F_i^{(j)}$ and p_i is a hash bit. Finally, n stego-images $I^{*(j)} = \{I^{*(1)}, I^{*(2)}, \ldots, I^{*(n)}\}$ with $2m \times 2m$ size are obtained until all pixels of the secret image are processed.

X_i' $x_i' = (x_{i8}x_{i7}x_{i6}x_{i5}x_{i4}x_{i3}F_{i8}F_{i7})_2$	W_i'' $w_i'' = (w_{i8}w_{i7}w_{i6}w_{i5}w_{i4}p_iF_{i6}F_{i5})_2$
V_i'' $v_i'' = (v_{i8}v_{i7}v_{i6}v_{i5}v_{i4}v_{i3}F_{i4}F_{i3})_2$	U_i' $u_i' = (u_{i8}u_{i7}u_{i6}u_{i5}u_{i4}u_{i3}F_{i2}F_{i1})_2$

Figure 2. The results of applying Yang *et al*.'s secret image sharing scheme to block $B_i^{(j)}$

2.2. Review of Chang *et al.*'s scheme

Chang *et al*. [7] employed all k secrets as coefficients in (k–1)-degree polynomial $F(x)$ to improve the stego-images quality. Only five bits $(x_{i8}, x_{i7}, x_{i6}, x_{i5}, x_{i4})_2$ in each block $B_i^{(j)}$ are used as $x_i^{(j)}$. Hence, the input of polynomial $x_i^{(j)}$ and the improved (k–1)-degree polynomial can be defined as following equations:

$$\begin{aligned} x_i &= (x_{i8}x_{i7}x_{i6}x_{i5}x_{i4})_2 \\ &= [(x_{i8}x_{i7}x_{i6}x_{i5}x_{i4}x_{i3}x_{i2}x_{i1})_2 \; AND \; 11111000_{(2)}] \,/\, 2^3 \end{aligned} \qquad (3)$$

$$F(x) = (s_1 + s_2 \times x^1 + \ldots + s_k \times x^{k-1}) \; mod \; 251, \qquad (4)$$

where x is the integer values, s_i={s_1, s_2, ⋯, s_k} are k integer secrets chosen from the secret image. However, this scheme is different from the other schemes since there is only one constant term secret as coefficient. [26]

For each unique value $x_i'^{(j)}$, the integer value of $F(x_i'^{(j)})$ is computed by Eq. (4) to form a secret share $F_i^{(1)}$, $F_i^{(2)}$, …, $F_i^{(n)}$, respectively. Each pair of ($x_i'^{(j)}$, $F(x_i'^{(j)})$) is a secret share of k secrets. After that, n secret shares $F_i^{(j)}$ are embedded into four pixels, X_i, W_i, V_i, and U_i, of each block $B_i^{(j)}$ by simple LSBs embedding method to form $B_i'^{(j)}$, respectively.

To prevent malicious participants and enhance authentication ability, a random bit stream generator with secret key K_2, six prime moduli $M_i^{(j)}$, and block index (i, j) are used to compute four authentication bits, p_{i4}, p_{i3}, p_{i2}, p_{i1} for each block $B_i'^{(j)}$. Next, n secret shares $F_i^{(j)}$ and authentication bits $p_i^{(j)}$ are embedded into four pixels X_i, W_i, V_i, and U_i of each block $B_i'^{(j)}$ by simple LSBs embedding method to form $B_i''^{(j)}$, respectively. Figure 3 shows the results of applying Chang *et al*.'s secret image sharing scheme with k integer secret pixels s_i={s_1, s_2, …, s_k} to block $B_i^{(j)}$, where F_{i8}, F_{i7}, …, F_{i1} are binary formats of the secret share $F_i^{(j)}$ and p_{i4}, p_{i3}, p_{i2}, p_{i1} are four authentication bits. Finally, n stego-images $I^{*(j)}$ = {$I^{*(1)}$, $I^{*(2)}$, …, $I^{*(n)}$} with $2m \times 2m$ size are obtained until all pixels of the secret image are processed.

X_i'' $x_i'' = (\underline{x_{i8}x_{i7}x_{i6}x_{i5}x_{i4}}\boxed{F_{i8}F_{i7}p_{i4}})_2$	W_i'' $w_i'' = (w_{i8}w_{i7}w_{i6}w_{i5}w_{i4}\boxed{F_{i6}F_{i5}p_{i3}})_2$
V_i'' $v_i'' = (v_{i8}v_{i7}v_{i6}v_{i5}v_{i4}\boxed{F_{i4}F_{i3}p_{i2}})_2$	U_i'' $u_i'' = (u_{i8}u_{i7}u_{i6}u_{i5}u_{i4}\boxed{F_{i2}F_{i1}p_{i1}})_2$

Figure 3. The results of applying Chang *et al.*'s scheme to block $B_i^{(j)}$

3. The proposed scheme

3.1. Secret Image Sharing Scheme

In order to enhance the image quality of the stego-image, the optimal LSBs method proposed by Chan and Cheng is adopted in our scheme. After applying the optimal pixel adjustment process (OPAP) to minimize the embedding error, the four pixels (X_i', W_i', V_i', and U_i') are modified to X_i'', W_i'', V_i'', and U_i'', respectively. The OPAP process can be briefly described as follows.

Let p_i, be the original pixel value of cover image, p_i' be the stego-pixel value after hiding the secret data by the simple LSB substitution scheme, and p_i'' be the refined stego-pixel value after applying the optimal LSBs method. The embedding error value δ_i between p_i and p_i' can be computed by $\delta_i = p_i' - p_i$. The basic idea of optimal LSBs

method is to minimize the distortion by adding or subtracting a 2^z factor from the embedded pixel, where z is the number of embedded bits. These adjustment operations do not affect the least z bits of the stego-image pixels. The optimal pixel adjustment process can be computed according to following rules:

$$p_i'' = \begin{cases} p_i' - 2^z, \text{ if } 2^{z-1} < \delta_i < 2^z \text{ and } p_i' \geq 2^z, \\ p_i' + 2^z, \text{ if } -2^z < \delta_i < -2^{z-1} \text{ and } p_i' < 256 - 2^z, \\ p_i', \text{ otherwise.} \end{cases} \quad (5)$$

The adjusted embedding error value δ_i' between p_i and p_i'' can be computed by $\delta_i' = p_i'' - p_i$. When apply the adjustment process, the absolute value of embedding error range is reduced from $0 \leq |\delta_i'| \leq 2^z - 1$ to $0 \leq |\delta_i'| \leq 2^{z-1}$. For more details about this method, please refer to Ref. [9, 11].

To prevent dishonest participants from executing malicious modification and enhance the authentication ability, a keyed-hash message authentication code [27] with secret key K_3, $H_{K3}(\cdot)$, block index B_{id} and stego-image identification $I_{id}^{(j)}$ are also used to compute the n hash bits $p_i^{(1)}, p_i^{(2)}, \ldots, p_i^{(n)}$ for each block $B_i^{(j)}$. The four hash bits p_{i4}, p_{i3}, p_{i2}, p_{i1} can be computed by the following equations:

$$\begin{aligned} h_i &= H_{K3}((X_i''-x_{i1}) \,||\, (W_i''-w_{i1}) \,||\, (V_i''-v_{i1}) \,||\, (U_i'' - u_{i1}) \,||\, B_{id} \,||\, I_{id}^{(j)})], \\ (p_{i4}, p_{i3}, p_{i2}, p_{i1}) &= (h_{i512}, h_{i511}, h_{i510}, h_{i509}) \oplus \cdots \oplus (h_{i4}, h_{i3}, h_{i2}, h_{i1}), \end{aligned} \quad (6)$$

where $(X_i''-x_{i1})$, $(W_i''-w_{i1})$, $(V_i''-v_{i1})$, and $(U_i'' - u_{i1})$ represent the $(x_{i8}x_{i7}x_{i6}x_{i5}x_{i4}F_{i8}F_{i7})$, $(w_{i8}w_{i7}w_{i6}w_{i5}w_{i4}F_{i6}F_{i5})$, $(v_{i8}v_{i7}v_{i6}v_{i5}v_{i4}F_{i4}F_{i3})$, and $(u_{i8}u_{i7}u_{i6}u_{i5}u_{i4}F_{i2}F_{i1})$; B_{id} is a block index, $B_{id} \in [1, m \times m]$; $I_{id}^{(j)}$ is a stego-image identification, $I_{id}^{(j)} \in [1, n]$; and $H_K(\cdot)$ is a standard hash function proposed in IETF RFC 2104 and FIPS 198 HMAC (Hash Message Authentication Code) [23, 28]. The fixed length hash message, take 512 bits for instance, h_i is calculated using a cryptographic hash function with a secret key K_3, and the Exclusive-OR operator symbol, $\oplus$, represents the binary operation on each bit. If any k watermarked stego-image has never suffered from malicious modification, then the secret image can be further recovered from the verified stego-image in the retrieval procedure. Otherwise, the malicious modification of the illegal stego-image will be detected and cannot pass the verification procedure.

Our method is also based on the (k, n)-threshold secret image sharing scheme proposed by Shamir. For the requirement of authentication capability, the dealer can choose from 0 to 4 pixel(s)'s LSBs to hide the authentication bit(s) into each 2 × 2 block in advance. In general, the more the authentication bits, the better the authentication capability. For convenience, only the secret image sharing scheme with four authentication bits is described as follows.

Because the gray value of a pixel is between 0 and 255, let the prime number pn equals to 251 which is the greatest prime number less than or equal to 2^8. Any secret pixels s_i larger than 250 in secret image will be modified to 250 to form s_i' by the following equation:

$$s_i' = \begin{cases} s_i, \text{ for } s_i \leq 250, \\ 250, \text{ for } s_i > 250, \end{cases} \quad (7)$$

where $1 \leqq i \leqq m \times m$.

To simplify the scheme, we use distinct image identification number $I_{id}^{(j)}$, e.g. 1, 2, 3, …, n, as the input of polynomial instead of $x_i^{(j)}$ for each cover image. Also, the conception and principle of one-way function [29-33] can be applied to avoid the security compromise of the same $I_{id}^{(j)}$. We can use it to compute $x_i^{(j)}$ from $x_{i-1}^{(j)}$ instead of the fixed $I_{id}^{(j)}$ both in the secret image sharing and retrieval processes. The initial values and one-way function are simply defined as follows: $\{x_0^{(1)}, x_0^{(2)}, \ldots, x_0^{(n)}\} = \{I_{id}^{(1)}, I_{id}^{(2)}, \ldots, I_{id}^{(n)}\}$ and $x_i^{(j)} = h(x_{i-1}^{(j)})$, where $i = 1$ to $m \times m$, $j = 1$ to n, and $0 < x_i^{(j)} < 251$. Table 1 is an example of one way function with $x_i^{(j)} = h(x_{i-1}^{(j)}) = (x_{i-1}^{(j)} \times 197) \bmod 251$. For all n integer values $x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(n)}, x_i^{(j)}$ must be distinct from the others to meet the requirement of Shamir's theorem.

Table 1. An example of one-way function

I_{id}	1	2	3	4	5	6	7	8	9
$i = 1$	$x_1^{(1)} = 1$	$x_1^{(2)} = 2$	$x_1^{(3)} = 3$	$x_1^{(4)} = 4$	$x_1^{(5)} = 5$	$x_1^{(6)} = 6$	$x_1^{(7)} = 7$	$x_1^{(8)} = 8$	$x_1^{(9)} = 9$
$i = 2$	$x_2^{(1)} = 197$	$x_2^{(2)} = 143$	$x_2^{(3)} = 89$	$x_2^{(4)} = 35$	$x_2^{(5)} = 232$	$x_2^{(6)} = 178$	$x_2^{(7)} = 124$	$x_2^{(8)} = 70$	$x_2^{(9)} = 16$
$i = 3$	$x_3^{(1)} = 155$	$x_3^{(2)} = 59$	$x_3^{(3)} = 214$	$x_3^{(4)} = 118$	$x_3^{(5)} = 22$	$x_3^{(6)} = 177$	$x_3^{(7)} = 81$	$x_3^{(8)} = 236$	$x_3^{(9)} = 140$
$i = 4$	$x_4^{(1)} = 164$	$x_4^{(2)} = 77$	$x_4^{(3)} = 241$	$x_4^{(4)} = 154$	$x_4^{(5)} = 67$	$x_4^{(6)} = 231$	$x_4^{(7)} = 144$	$x_4^{(8)} = 57$	$x_4^{(9)} = 221$
$i = 5$	$x_5^{(1)} = 180$	$x_5^{(2)} = 109$	$x_5^{(3)} = 38$	$x_5^{(4)} = 218$	$x_5^{(5)} = 147$	$x_5^{(6)} = 76$	$x_5^{(7)} = 5$	$x_5^{(8)} = 185$	$x_5^{(9)} = 114$
$i = 6$	$x_6^{(1)} = 69$	$x_6^{(2)} = 138$	$x_6^{(3)} = 207$	$x_6^{(4)} = 25$	$x_6^{(5)} = 94$	$x_6^{(6)} = 163$	$x_6^{(7)} = 232$	$x_6^{(8)} = 50$	$x_6^{(9)} = 119$
					…				
$i = m \times m$	$x_{m\times m}^{(1)} = 83$	$x_{m\times m}^{(2)} = 166$	$x_{m\times m}^{(3)} = 249$	$x_{m\times m}^{(4)} = 81$	$x_{m\times m}^{(5)} = 164$	$x_{m\times m}^{(6)} = 247$	$x_{m\times m}^{(7)} = 79$	$x_{m\times m}^{(8)} = 162$	$x_{m\times m}^{(9)} = 245$

We also utilize all k coefficients in (k–1)-degree polynomial to share k secret pixels for each 2 × 2 block. Hence, the (k–1)-degree polynomial can be defined as following equation:

$$F(x_i) = (s_1 + s_2 \times x_i^1 + \ldots + s_k \times x_i^{k-1}) \bmod pn, \qquad (8)$$

where x_i is the input of polynomial, $s = \{s_1, s_2, \cdots, s_k\}$ are k integer secrets chosen from the secret image, the prime number pn is equal to 251, the modulo operation symbol, mod, represents the remainder after integer division, value of x_i and $s_1, s_2, \ldots, s_k$ are between 0 to 250. For each unique $x_i^{(j)}$, the integer value of $F(x_i^{(j)})$ is computed by Eq. (8) to form a secret share $F_i^{(1)}, F_i^{(2)}, \ldots, F_i^{(n)}$, respectively. Each pair of $(x_i^{(j)}, F(x_i^{(j)}))$ is a secret share of k secrets.

In general, n secret shares $F_i^{(j)}$ and authentication bits of previous block $B_{i-1}^{(j)}$ are embedded into four pixels X_i, W_i, V_i, and U_i of each block $B_i^{(j)}$ by simple LSBs embedding method to form $B_i'^{(j)}$, respectively. The result of X_i', W_i', V_i', and U_i' in each block $B_i'^{(j)}$ can be computed by the following equation:

$$\begin{aligned} X_i' &= X_i - (X_i \bmod 2^3) + (F_{i8}F_{i7})_2 \times 2 + p_{i-1,4}, \\ W_i' &= W_i - (W_i \bmod 2^3) + (F_{i6}F_{i5})_2 \times 2 + p_{i-1,3}, \\ V_i' &= V_i - (V_i' \bmod 2^3) + (F_{i4}F_{i3})_2 \times 2 + p_{i-1,2}, \\ U_i' &= U_i - (U_i \bmod 2^3) + (F_{i2}F_{i1})_2 \times 2 + p_{i-1,1}, \end{aligned} \qquad (9)$$

where the initial value of $\{p_{i-1,4}, p_{i-1,3}, p_{i-1,2}, p_{i-1,1}\}$ are {0, 0, 0, 0} for the first block (at the top-left corner) of each cover image. Every block contains four hash bits of the previous block. And the OPAP is executed after hiding the secret share and authentication bits. Only the first block of cover image is an exception in which some LSBs will be modified after authentication phase. The four authentication bits of the last block (at the bottom-right corner) are computed and embedded into the first block in the end.

Figure 4 shows the results of applying our secret image sharing scheme with k secret pixels to block $B_i^{(j)}$, where F_{i8}, F_{i7}, …, F_{i1} are binary formats of the secret share $F_i^{(j)}$ and $p_{i-1,4}$, $p_{i-1,3}$, $p_{i-1,2}$, $p_{i-1,1}$ are four authentication bits of the previous block. Finally, n stego-images $I^{*(j)} = \{I^{*(1)}, I^{*(2)}, \ldots, I^{*(n)}\}$ with $2m \times 2m$ size are obtained until all pixels of the secret image are processed.

$X_i^{''}$ $x_i^{''}=(x_{i8}x_{i7}x_{i6}x_{i5}x_{i4}\boxed{F_{i8}F_{i7}p_{i-1,4}})_2$	$W_i^{''}$ $w_i^{''}=(w_{i8}w_{i7}w_{i6}w_{i5}w_{i4}\boxed{F_{i6}F_{i5}p_{i-1,3}})_2$
$V_i^{''}$ $v_i^{''}=(v_{i8}v_{i7}v_{i6}v_{i5}v_{i4}\boxed{F_{i4}F_{i3}p_{i-1,2}})_2$	$U_i^{''}$ $u_i^{''}=(u_{i8}u_{i7}u_{i6}u_{i5}u_{i4}\boxed{F_{i2}F_{i1}p_{i-1,1}})_2$

Figure 4. The results of applying our proposed scheme with four authentication bits to block $B_i^{(j)}$

3.2. Secret Image Sharing and Retrieval Algorithms

The following is our secret image sharing and secret image retrieval algorithms with four authentication bits.

Algorithm 1: **Proposed novel secret image sharing procedure**

Input: A secret image $S = \{s_1, s_2, \ldots, s_{m\times m}\}$, n cover images $I^{(j)}=\{I^{(1)}, I^{(2)}, \ldots, I^{(n)}\}$, and a secret key K_3

Output: n stego-images $I^{*(j)} = \{I^{*(1)}, I^{*(2)}, \ldots, I^{*(n)}\}$

Step 1. Divide n cover images $I^{(1)}, I^{(2)}, \ldots, I^{(n)}$ into $m \times m$ non-overlapping 2×2 blocks $B_i^{(j)}$, where $i = 1$ to $m \times m$ and $j = 1$ to n.

Step 2. Choose a set of k pixel values s_i within 0 to 250 from a secret image S used as the secret $s_i = \{s_1, s_2, \ldots, s_k\}$ by Eq. (7).

Step 3. For each distinct number $x_i^{(j)}$ as the input of polynomial, compute the integer value of $F(x_i^{(j)})$ by Eq. (8) to form a secret share $F_i^{(1)}, F_i^{(2)}, \ldots, F_i^{(n)}$, respectively.

Step 4. The initial values of $\{p_{i-1,4}, p_{i-1,3}, p_{i-1,2}, p_{i-1,1}\}$ are $\{0, 0, 0, 0\}$. Except for the first block at the top-left corner, compute the previous block's authentication bits, $p_{i-1,4}$, $p_{i-1,3}$, $p_{i-1,2}$, and $p_{i-1,1}$, in each block $B_i^{'(j)}$ by Eq. (6).

Step 5. Use simple LSBs method to hide the secret share $F_i^{(j)}$ and authentication bits $p_{i-1,4}$, $p_{i-1,3}$, $p_{i-1,2}$, and $p_{i-1,1}$ in binary format into each block $B_i^{(j)}$ to form $B_i^{'(j)}$ by Eq. (9), where $i = 1$ to $m \times m$ and $j = 1$ to n.

Step 6. Apply an optimal pixel adjustment process (OPAP) [9, 11, 12] to each stego-block $B_i^{'(j)}$ by Eq. (5), respectively.

Step 7. Repeat Step 2 to Step 6 until all pixels of the secret image are processed.

Step 8. For each stego image, the four authentication bits of the last block are calculated and embedded into the first block in the end of process.

Algorithm 2: **Proposed novel secret image retrieval procedure**

Input: n stego-images $I^{*(j)} = \{I^{*(1)}, I^{*(2)}, \ldots, I^{*(n)}\}$, and a secret key K_3

Output: A secret image $S = \{s_1, s_2, \ldots, s_{m\times m}\}$

Step 1. Divide n stego-images $I^{*(j)}=\{I^{*(1)}, I^{*(2)}, \ldots, I^{*(n)}\}$ into $m \times m$ nonoverlapping 2×2 blocks $B_i^{*(j)}$, where $i = 1$ to $m \times m$ and $j = 1$ to n.

Step 2. The authentication bits are extracted from the first block at the top-left corner only for the last block at the bottom-right corner. Except for the last block, it is easy to extract the authentication bits p_{i4}^*, p_{i3}^*, p_{i2}^*, p_{i1}^* from LSBs of each stego-block $B_{i+1}^{*(j)}$.

Step 3. Calculate and compare the authentication bits p_{i4}, p_{i3}, p_{i2}, p_{i1} in each stego-block $B_i^{*(j)}$ by Eq. (6), respectively.

Step 4. If $p_i^{(j)}$ is equal to $p_i^{*(j)}$, then ith share is assumed to be authentic and proceed to step 5. Otherwise, the block has been modified and the image is flagged as illegal.

Step 5. Use simple LSBs method to extract $F_i^{*(j)}$ from each stego-block $B_i^{*(j)}$, where $i = 1$ to $m \times m$ and $j = 1$ to n.

Step 6. The 8-bits $F_i^{*(j)}$ values from k stego-blocks and distinct number $x_i^{(j)}$ are substituted in the Lagrange's interpolation formula to retrieve the corresponding secret pixels s_1^*, s_2^*, …, s_k^*.

Step 7. Repeat Step 2 to Step 6 until all pixels of the secret image are obtained.

3.3. An Example of (2, 3)-Threshold Procedure

121 $(01111001)_2$	231 $(11100111)_2$	173 $(10101101)_2$	102 $(01100110)_2$	218 $(11011010)_2$	153 $(10011001)_2$
103 $(01100111)_2$	169 $(10101001)_2$	149 $(10010101)_2$	207 $(11001111)_2$	187 $(10111011)_2$	106 $(01101010)_2$

(a) 2×2 original cover blocks

126 $(01111111)_2$	224 $(11100001)_2$	170 $(10101011)_2$	96 $(01100000)_2$	220 $(11011100)_2$	158 $(10011111)_2$
100 $(01100101)_2$	174 $(10101111)_2$	146 $(10010011)_2$	202 $(11001011)_2$	188 $(10111101)_2$	108 $(01101100)_2$

(b) Stego-blocks obtained by the LSBs method

127 $(01111111)_2$	224 $(11100000)_2$	170 $(10101010)_2$	97 $(01100001)_2$	221 $(11011101)_2$	159 $(10011111)_2$
101 $(01100101)_2$	174 $(10101110)_2$	147 $(10010011)_2$	202 $(11001010)_2$	188 $(10111100)_2$	109 $(01101101)_2$

(c) The results of stego-blocks after embedding authentication bits

119 (-2^3) $(01110111)_2$	232 $(+2^3)$ $(11101000)_2$	170 $(10101010)_2$	105 $(+2^3)$ $(01101001)_2$	221 $(11011101)_2$	151 (-2^3) $(10010111)_2$
101 $(01100101)_2$	166 (-2^3) $(10100110)_2$	147 $(10010011)_2$	210 $(+2^3)$ $(11010010)_2$	188 $(10111100)_2$	109 $(01101101)_2$

(d) Refined stego-blocks after the OPAP process

Figure 5. An example of our proposed scheme with four authentication bits

An example applying our (2, 3)-threshold secret image sharing procedure is shown in Figure 5. Figure 5 (a) shows the three 2×2 original cover blocks. Assume that secret $\{s_1, s_2\} = \{86, 117\}$ and $\{I_{id}^{(1)}, I_{id}^{(2)}, I_{id}^{(3)}\} = \{1, 2, 3\}$, then the three secret shares are $F(1) = 86+117\times1 \pmod{251} = 203_{(10)} = 11001011_{(2)}$, $F(2) = 86+117\times2 \pmod{251} = 69_{(10)}$

$=01000101_{(2)}$, and $F(3) = 86+117\times3 \pmod{251} = 186_{(10)} = 10111010_{(2)}$. After embedding the secret shares into cover blocks, the three stego-blocks obtained by the simple LSBs substitution method are shown in Figure 5 (b). Suppose the authentication bits of previous block are computed by Eq. (6), for example: $1010_{(2)}$, $0110_{(2)}$, and $1101_{(2)}$. And the four bits for identity authentication will be inserted in the least significant bit of stego-blocks. The results of stego-blocks after embedding authentication bits, $1010_{(2)}$, $0110_{(2)}$, and $1101_{(2)}$, are shown in Figure 5 (c). Then, the distortion between the original cover blocks and stego-blocks can be computed as follows: (121−127 = -6, 231−224 = 7, 103−101 = 2, 169−174 = -5), (173−170 = 3, 102−97 = 5, 149−147 = 2, 207−202 = 5), and (218−221 = -3, 153−159 = -6, 187−188 = -1, 106−109 = -3). Figure 5 (d) shows the refined stego-blocks after the OPAP process by Eq. (5). Next, the distortion between the original cover blocks and refined stego-blocks can be computed as follows: (121−119 = 2, 231−232 = -1, 103−101 = 2, 169−166 = 3), (173−170 = 3, 102−105 = -3, 149−147 = 2, 207−210 = -3), and (218−221 = -3, 153−151 = 2, 187−188 = -1, 106−109 = -3). As you can see, the distortion between original cover blocks and stego-blocks is from (-6, 7, 2, -5), (3, 6, 2, 5), and (-3, -6, -1, -3) reduced to (2, -1, 2, 3), (3, -3, 2, -3), and (-3, 2, -1, -3). From this example, we can clearly observe that the distortion of the stego-image can be greatly reduced by using the optimal LSBs method.

4. Experimental results

The algorithms were implemented in Java programming language on a PC with Intel Core 2 Duo 2.4 GHz CPU, 1024 MB RAM, and Windows XP Professional system. The six standard grayscale test images (General test pattern, Airplane, Baboon, Lena, Pepper, and Sailboat) were delivered from USC-SIPI image database[34]. To compare the performance of our algorithm with Lin and Tsai's, Yang *et al*.'s, and Chang *et al*.'s methods, the experiments were carried out on the same images. Let's suppose that the secret image is "General test pattern" with 256×256 pixels, as shown in Figure 6 (a). Five 512×512 pixels images "Airplane", "Baboon", "Lena", "Pepper", and "Sailboat" are used as the cover images. Each of these cover images is shown in Figure 6 (b) – (f), respectively. After applying (2, *n*)-threshold scheme, Figure 7, Figure 8, and Figure 9 show the Lin and Tsai's, Yang *et al*.'s, and Chang *et al*.'s results, with our results shown in Figure 10 (a) – (e), respectively.

To evaluate the visual quality of stego images by using the human eye, we enlarged partial area of original cover images and stego-images, as shown in left column and right column of Figure 11. Figure 11 (a), (c), and (e) show the cropped area in the original Lena, Pepper, and Sailboat images and Figure 11 (b), (d), and (f) show the cropped area in the stego-images of Lena, Pepper, and Sailboat. The distortion between original cover images and stego-images is visually almost imperceptible from visual perception as demonstrated in Figure 11.

Two important measures of a secret image sharing scheme are evaluated to demonstrate the superiority of the proposed scheme quantitatively compared with existing methods reported in the literature. The first factor for quantitative comparison is to measure the distortion between an original cover image and the stego-image. The lower distortion indicates the better visual image quality of the stego-image. Enhancing the visual image quality and authentication capability of stego-image are new challenges for researchers working on development of novel secret image sharing scheme.

Table 1 contains our scheme's image quality results compared with Lin and Tsai's, Yang *et al*.'s, and Chang *et al*.'s schemes after applying (2, *n*)-threshold scheme with the same payload capacity (65,536 pixels). The peak signal-to-noise ratio (PSNR) is commonly used as a measure of image quality since it is easily defined via the mean squared error (MSE). The formulas of PSNR and MSE calculation are defined below:

$$\text{PSNR} = 10 \times \log_{10} \frac{255^2}{\text{MSE}} \text{dB}; \tag{10}$$

$$\text{MSE} = \frac{\sum_{x=1}^{M} \sum_{y=1}^{N} \left[P(x, y) - P^*(x, y) \right]^2}{M \times N}, \tag{11}$$

where M and N represent the image size, $P(x, y)$ and $P^*(x, y)$ stand for the original pixel value and stego-pixel value in position (x, y). The greater the PSNR value, the less the distortion of image will be. In other words, the larger the PSNR value the smaller the possibility of a variety of visual attacks by human eye.

In Lin and Tsai's and Yang *et al*.'s schemes, only 1 bit for authentication in each 2×2 block. The proposed and Chang *et al*.'s schemes are presented with 4 bits for authentication in each 2×2 block. Furthermore, the authentication capability to identify tampered blocks among Lin and Tsai's, Yang *et al*.'s and Chang *et al*.'s schemes are demonstrated in [7]. The average PSNR value of the stego-images in Lin and Tsai's, Yang *et al*.'s, Chang *et al*.'s and the proposed methods are 39.19, 41.58, 39.84, and 43.54, respectively. Although Chang *et al*.'s average PSNR value is smaller than Yang *et al*.'s. Nevertheless, Chang *et al*.'s scheme has strong authentication capability. The result of image quality analysing shows that our scheme indeed has a better image quality and really enhances the quality of stego-image than other methods. According to the experimental results, the average PSNR value of the stego-images by our proposed scheme is 43.54 dB, which clearly outperformed the lately related research results.

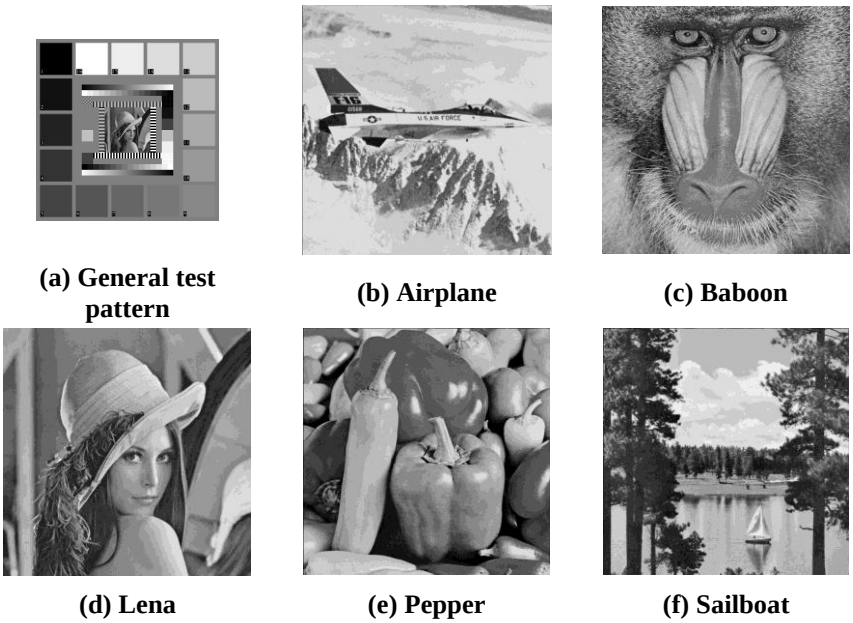
In addition, another quantitative measure is also performed to estimate authentication capability of the proposed scheme under various authentication bits conditions. The probability of authenticating a fake or counterfeit stego-block indicates the effectiveness of the proposed authentication method. The criterion for integrity verification is measured by detection ratio (DR). The *DR* means the detection ratio against the tampered region, and it is defined as the following equation:

$$DR = \frac{NTPD}{NTP}, \tag{12}$$

where *NTP* and *NTPD* represent the number of the tampered pixels and the number of the tampered pixels that are detected respectively. [7]

To estimate the capability of identifying the tampered region under various authentication bits conditions. Figure 12 (a) and Figure 13 (a) show the tampered stego-images, Airplane and Lena, from Figure 10 (a) and (b). The forged stego-images, House and Chemical Plant, are also shown in Figure 14 (a) and Figure 15 (a). The results are shown in Figure 12 (b) – (e), Figure 13 (b) – (e), Figure 14 (b) – (e), and Figure 15 (b) – (e), respectively. According to the experimental results, the average detection ratios *DRs* under various authentication bits conditions are 0.95, 0.89, 0.76, and 0.5, respectively. Furthermore, the experimental results show that the detection ratios in Lin and Tsai's, Yang *et al*.'s, and Chang *et al*.'s schemes are 0, 0.52 and 0.97 in Ref [7]. In summary, the authentication ability of

the proposed scheme can detect a fake stego-block with a high probability 0.95 as compared with Lin and Tsai's, Yang *et al.*'s, and Chang *et al.*'s schemes.



(a) General test pattern **(b) Airplane** **(c) Baboon**

(d) Lena **(e) Pepper** **(f) Sailboat**

Figure 6. (a) Secret image *S*; (b)–(f) Cover images $I^{(j)}$

Table 2. The PSNR and MSE of stego-images $I^{*(j)}$ with the same payload capacity (The unit of PSNR is dB)

Secret Image (256×256)	Stego-Image (512×512)	Lin-Tasi Scheme		Yang *et al.*'s Scheme		Chang *et al.*'s Scheme		Our Scheme	
		MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR
General test pattern	Airplane	7.74	39.25	4.43	41.66	5.49	40.73	2.89	43.53
	Baboon	7.85	39.18	4.55	41.55	6.59	39.94	2.88	43.54
	Lena	7.80	39.20	4.50	41.60	5.98	40.37	2.88	43.54
	Pepper	7.86	39.17	4.54	41.56	7.63	39.30	2.86	43.56
	Sailboat	7.89	39.16	4.59	41.51	8.45	38.86	2.87	43.55
Average		7.83	39.19	4.52	41.58	6.83	39.84	2.88	43.54

(a) 39.25 dB **(b) 39.18 dB** **(c) 39.20 dB** **(d) 39.17 dB** **(e) 39.16 dB**

Figure 7. Stego-images $I^{*(j)}$ of Lin and Tsai scheme

(a) 41.66 dB **(b) 41.55 dB** **(c) 41.60 dB** **(d) 41.56 dB** **(e) 41.51 dB**

Figure 8. Stego-images $I^{*(j)}$ of Yang *et al.*'s scheme

(a) 40.73 dB **(b) 39.94 dB** **(c) 40.37 dB** **(d) 39.30 dB** **(e) 38.86 dB**

Figure 9. Stego-images $I^{*(j)}$ of Chang *et al.*'s scheme

(a) 43.53 dB **(b) 43.54 dB** **(c) 43.54 dB** **(d) 43.56 dB** **(e) 43.55 dB**

Figure 10. Stego-images $I^{*(j)}$ of our scheme

(a) Enlarged cover image Lena from Figure 6 (d)

(b) Enlarged stego-image Lena from Figure 10 (c)

(c) Enlarged cover image Pepper from Figure 6 (e)

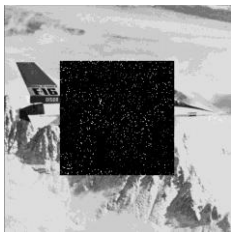
(d) Enlarged stego-image Pepper from Figure 10 (d)

(e) Enlarged cover image Sailboat from Figure 6 (f)

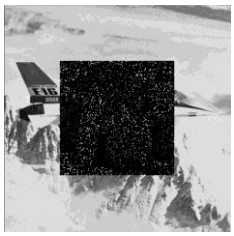
(f) Enlarged stego-image Sailboat from Figure 10 (e)

Figure 11. Enlarged partial area of the original cover images and stego-images

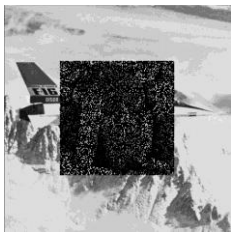
(a) Tampered stego-image Airplane from Figure 10 (a)



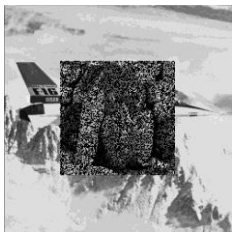
(b) 4 authentication bits (DR = 0.95)



(c) 3 authentication bits (DR = 0.89)

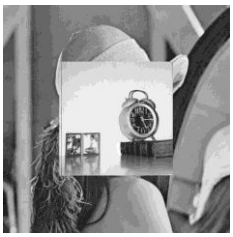


(d) 2 authentication bits (DR = 0.76)

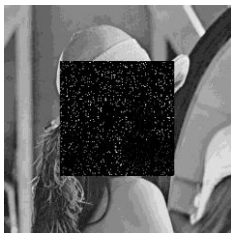


(e) 1 authentication bit (DR = 0.51)

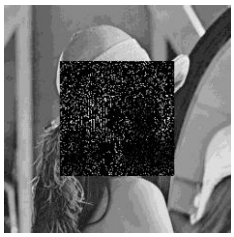
Figure 12. The capability of the proposed authentication method



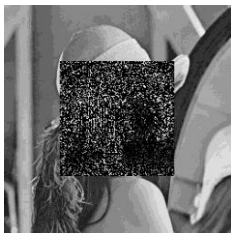
(a) Tampered stego-image Lena from Figure 10 (b)



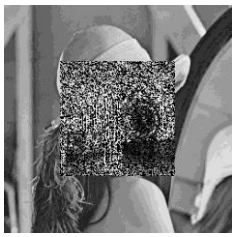
(b) 4 authentication bits (DR = 0.95)



(c) 3 authentication bits (DR = 0.88)



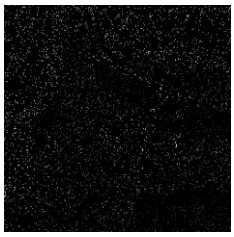
(d) 2 authentication bits (DR = 0.76)



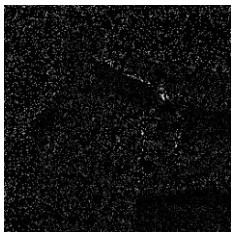
(e) 1 authentication bit (DR = 0.51)

Figure 13. The capability of the proposed authentication method

(a) Forged stego-image House



(b) 4 authentication bits (DR = 0.94)



(c) 3 authentication bits (DR = 0.89)

(d) 2 authentication bits (DR = 0.77)

(e) 1 authentication bit (DR = 0.48)

Figure 14. The capability of the proposed authentication method

(a) Forged stego-image Chemical Plant

(b) 4 authentication bits (DR = 0.94)

(c) 3 authentication bits (DR = 0.88)

(d) 2 authentication bits (DR = 0.75)

(e) 1 authentication bit (DR = 0.50)

Figure 15. The capability of the proposed authentication method

5. Discussion and Conclusion

In the literature, almost all the recent secret image sharing schemes are block-wise and designed in the spatial domain. For this reason, gray scale (or gray level) images are widely used for hiding data. In general, these schemes divide cover images into non-overlapping 2 × 2 blocks and directly replace the least significant bits of pixels in each bock with the secret share and authentication code. The replacing process may introduce some distortion or artifacts in the stego-image. We introduce one of the improved methods called the optimal LSBs scheme in this manuscript. The method can greatly improve the image quality by applying an optimal pixel adjustment process to the stego-image.

Suppose that all the pixels in each 2 × 2 block of the cover image are used for the embedding of secret share and authentication code by the simple LSB substitution method. Theoretically, the absolute value of the embedding error range is $0 \le |\delta_i| \le 2^z - 1$. And the worst mean-square-error (WMSE) between the cover-image and the stego-image is $(2^3-1)^2 = 49$ in this phase. [9, 11] Accordingly, the worst case PSNR_{worst} of the stego-image can be computed by the following equation:

$$\text{PSNR}_{worst} = 10 \times \log_{10} \frac{255^2}{\text{WMSE}} \text{dB} = 10 \times \log_{10} \frac{255^2}{(2^3-1)^2} \text{dB} = 31.23 \text{ dB}.$$

The authentication phase involves embedding the watermark by replacing the least-significant-bit of each pixel with a bit of the authentication code before the optimal pixel adjustment process. In our scheme, with four authentication bits condition, the LSB of each pixel is used to embed the watermark into each 2 × 2 block. The absolute value of the embedding error range is only $0 \le |\delta_i'| \le 2^{z-1}$. Let WMSE* be the worst mean-square-error between the cover image and the stego-image obtained by OPAP. The result value of WMSE* can be obtained as 16. Accordingly, the worst case PSNR_{worst*} of the stego-image can be computed by the following equation:

$$\text{PSNR}_{worst*} = 10 \times \log_{10} \frac{255^2}{\text{WMSE*}} \text{dB} = 10 \times \log_{10} \frac{255^2}{(2^{3-1})^2} \text{dB} = 36.09 \text{ dB}.$$

If the OPAP is adopted, the WMSE* can be reduced from 49 to 16 and the PSNR_{worst*} is improved from 31.23 to 36.09 dB. It is clearly demonstrated that the gain of PSNR is about 4.86 dB in the worst case.

Table 3 shows Lin and Tsai's, Yang *et al.*'s, and Chang *et al.*'s schemes simply to be compared with our scheme's image quality results in worst case under different payload capacity and various authentication bits conditions. All k coefficients in (k–1)-degree polynomial $F(x)$ are used to share k secret pixels in Chang *et al.* and our schemes. In other words, the maximum embedding capacities with respect to the cover image size and the factor k in (k, n)-threshold sharing scheme. In one authentication bit condition, the worst PSNR value of the stego-images in Lin and Tsai's, Yang *et al.*'s, and the proposed methods are 32.48, 35.34, and 39.68. While in four bits for identity authentication condition, the worst PSNR value of the stego-images in Chang *et al.*'s and the proposed methods are 31.23 and 36.09. The result of worst peak signal-to-noise ratio analyzing shows that our scheme indeed has better image quality than the others while the same authentication bits. In this study, several performance features are

compared and shown Table 4. From the table, we can easily observe that the average PSNR values of the stego-images by our proposed scheme are around 43.54 dB, which clearly outperforms three other existing approaches.

Table 3. Evaluation of stego-image quality in the worst case under different conditions

	Worst Mean-Square-Error (MSE_{worst})	Worst Peak Signal-to-Noise Ratio ($PSNR_{worst}$)	Authentication Code	Maximum Capacity (Pixels)
Lin and Tsai's Scheme	$\frac{\sum_{x=1}^{256}\sum_{y=1}^{256}\left[0^2+(2^3-1)^2+(2^3-1)^2+(2^3-1)^2\right]}{512\times512}=36.75$	$10\times\log_{10}\frac{255^2}{36.75}=32.48$ dB	1 bit	256×256 = 65,536
Yang *et al.*'s Scheme	$\frac{\sum_{x=1}^{256}\sum_{y=1}^{256}\left[(2^2-1)^2+(2^3-1)^2+(2^2-1)^2+(2^2-1)^2\right]}{512\times512}=19.00$	$10\times\log\frac{255^2}{19.00}=35.34$ dB	1 bit	256×256 = 65,536
Chang *et al.*'s Scheme	$\frac{\sum_{x=1}^{256}\sum_{y=1}^{256}\left[(2^3-1)^2+(2^3-1)^2+(2^3-1)^2+(2^3-1)^2\right]}{512\times512}=49.00$	$10\times\log_{10}\frac{255^2}{49.00}=31.23$ dB	4 bits	k×256×256 = k × 65,536
Our Scheme	$\frac{\sum_{x=1}^{256}\sum_{y=1}^{256}\left[(2^2)^2+(2)^2+(2)^2+(2)^2\right]}{512\times512}=7.00$	$10\times\log_{10}\frac{255^2}{7.00}=39.68$ dB	1 bit	k×256×256 = k × 65,536
	$\frac{\sum_{x=1}^{256}\sum_{y=1}^{256}\left[(2^2)^2+(2^2)^2+(2)^2+(2)^2\right]}{512\times512}=10.00$	$10\times\log_{10}\frac{255^2}{10.00}=38.13$ dB	2 bits	k×256×256 = k × 65,536
	$\frac{\sum_{x=1}^{256}\sum_{y=1}^{256}\left[(2^2)^2+(2^2)^2+(2^2)^2+(2)^2\right]}{512\times512}=13.00$	$10\times\log_{10}\frac{255^2}{13.00}=38.13$ dB	3 bits	k×256×256 = k × 65,536
	$\frac{\sum_{x=1}^{256}\sum_{y=1}^{256}\left[(2^2)^2+(2^2)^2+(2^2)^2+(2^2)^2\right]}{512\times512}=16.00$	$10\times\log_{10}\frac{255^2}{16.00}=36.09$ dB	4 bits	k×256×256 = k × 65,536

Table 4. Feature comparisons among Lin and Tsai's, Yang *et al.*'s, Chang *et al.*'s and our scheme

Feature	Lin and Tsai [5]	Yang *et al.* [6]	Chang *et al.* [7]	Our Scheme
Secret Image Sharing	Yes	Yes	Yes	Yes
Meaningful Stego-image	Yes	Yes	Yes	Yes
Image Quality of Stego-Image	39.19 dB	41.58 dB	39.84 dB	43.54 dB
Authentication Code	1 bit	1 bit	4 bits	4 bits
Maximum Capacity (Pixels)	65,536	65,536	k × 65,536	k × 65,536

Note: The symbol k is the factor in (k, n)-threshold sharing scheme.

Recently, Eslami and Ahmadabadi [35] proposed a dynamic embedding and authentication chaining scheme in 2011. They utilize all k coefficients in $f(x)$ to share k secret pixels and arrange the positions of 10 bits (8 bits secret share plus 2 authentication bits) for (2, n) and (3, n)-threshold schemes in each consecutive 8 and 11 pixels block, respectively. The results of applying Eslami and Ahmadabadi (2, n) and (3, n)-threshold schemes with two authentication bits are shown in Figure 16 (a) and (b).

The proposed scheme also can extend to $k \times 2 \times 2$ pixels block method under the constant payload capacity. We can rearrange the positions of 12 bits (8 bits secret share plus 4 authentication bits) in each k consecutive 2×2 block to achieve an objective of high image quality. Figure 17 (a) and (b) show the results of stego blocks by applying our extended (2, n) and (3, n)-threshold schemes with four authentication bits.

Table 5 shows the comparison of the image quality in the worst case between our extended scheme and Eslami and Ahmadabadi's scheme by using (2, n), (3, n), and (4, n)-threshold schemes respectively. Clearly, the $PSNR_{worst}$ value of our extended scheme is very close to Eslami and Ahmadabadi's scheme. By using (2, n), (3, n), and (4, n)-threshold schemes, the $PSNR_{worst}$ values of the stego-images in Eslami and Ahmadabadi's and the proposed methods are {43.36 dB, 48.54 dB, 50.17 dB} and {44.15 dB, 48.13 dB, 49.38 dB}, respectively. Obviously, the difference between the Eslami and Ahmadabadi's and the extended schemes is only about 0.79 dB which is hardly noticeable to the human eye.

As described above, the proposed scheme utilizes all k coefficients in $f(x)$ to share k secret pixels for each 2×2 block. The average PSNR value is around 43.54 dB, and the maximum capacity is $k \times 256 \times 256$ pixels. By contrast, the PSNR value of Eslami and Ahmadabadi's scheme is large than 48.10 dB, but the maximum capacity is only about 256×256 pixels. Moreover, the extended scheme is further proposed to achieve image quality as high as that in Eslami and Ahmadabadi's scheme.

In this manuscript, we have proposed an improvement for the flaw in Yang *et al.*'s and Chang *et al.*'s schemes to enhance the stego-image quality. The evaluation of the results supports the claim that our scheme has significantly better result than the others. Several experimental results are also provided to demonstrate the efficacy of the authentication capability of the proposed scheme under various authentication bits conditions. The results and discussion clearly indicate that the proposed method achieves both high visual image quality and high authentication capability of stego-image.

However, the common problem is that the stego-image cannot be recovered to the original cover image state. Further studies will be proposed on the lossless image sharing scheme for secret image and stego-images. The goal is to construct the lossless secret image from stego-images and increase the ability to recover the stego-images back to the original cover image.

B_1	B_2	B_3	B_4	B_5	B_6	B_7	B_8
$(b_8^1 \ldots b_2^1 \boxed{b_1^1})_2$	$(b_8^2 \ldots b_2^2 \boxed{b_1^2})_2$	$(b_8^3 \ldots b_2^3 \boxed{b_1^3})_2$	$(b_8^4 \ldots b_2^4 \boxed{b_1^4})_2$	$(b_8^5 \ldots b_2^5 \boxed{b_1^5})_2$	$(b_8^6 \ldots b_2^6 \boxed{b_1^6})_2$	$(b_8^7 \ldots \boxed{b_2^7 b_1^7})_2$	$(b_8^8 \ldots \boxed{b_2^8 b_1^8})_2$

(a) (2, *n*)-threshold scheme

B_1	B_2	B_3	B_4	B_5	B_6	B_7	B_8	B_9	B_{10}	B_{11}
$(b_8^1 \ldots \boxed{b_1^1})_2$	$(b_8^2 \ldots \boxed{b_1^2})_2$	$(b_8^3 \ldots \boxed{b_1^3})_2$	$(b_8^4 \ldots \boxed{b_1^4})_2$	$(b_8^5 \ldots \boxed{b_1^5})_2$	$(b_8^6 \ldots \boxed{b_1^6})_2$	$(b_8^7 \ldots \boxed{b_1^7})_2$	$(b_8^8 \ldots \boxed{b_1^8})_2$	$(b_8^9 \ldots \boxed{b_1^9})_2$	$(b_8^{10} \ldots \boxed{b_1^{10}})_2$	$(b_8^{11} \ldots b_1^{11})_2$

(b) (3, *n*)-threshold scheme

Figure 16. The results of applying Eslami and Ahmadabadi's scheme with two authentication bits

X_i''	W_i''	X_{i+1}''	W_{i+1}''
$(x_8 \ldots x_2 \boxed{F_{i8}})_2$	$(w_8 \ldots w_2 \boxed{F_{i7}})_2$	$(x_8 \ldots x_3 \boxed{F_{i4} p_{i-1,4}})_2$	$(w_8 \ldots w_3 \boxed{F_{i3} p_{i-1,3}})_2$
V_i''	U_i''	V_{i+1}''	U_{i+1}''
$(v_8 \ldots v_{i,2} \boxed{F_{i6}})_2$	$(u_8 \ldots u_2 \boxed{F_{i5}})_2$	$(v_8 \ldots v_3 \boxed{F_{i2} p_{i-1,2}})_2$	$(u_8 \ldots u_3 \boxed{F_{i1} p_{i-1,1}})_2$

(a) (2, *n*)-threshold scheme

X_i''	W_i''	X_{i+1}''	W_{i+1}''	X_{i+2}''	W_{i+2}''
$(x_8 \ldots x_2 \boxed{F_{i8}})_2$	$(w_8 \ldots w_2 \boxed{F_{i7}})_2$	$(x_8 \ldots x_2 \boxed{F_{i4}})_2$	$(w_8 \ldots w_2 \boxed{F_{i3}})_2$	$(x_8 \ldots x_2 \boxed{p_{i-1,4}})_2$	$(w_8 \ldots w_2 \boxed{p_{i-1,3}})_2$
V_i''	U_i''	V_{i+1}''	U_{i+1}''	V_{i+2}''	U_{i+2}''
$(v_8 \ldots v_{i,2} \boxed{F_{i6}})_2$	$(u_8 \ldots u_2 \boxed{F_{i5}})_2$	$(v_8 \ldots v_2 \boxed{F_{i2}})_2$	$(u_8 \ldots u_2 \boxed{F_{i1}})_2$	$(v_8 \ldots v_2 \boxed{p_{i-1,2}})_2$	$(u_8 \ldots u_2 \boxed{p_{i-1,1}})_2$

(b) (3, *n*)-threshold scheme

Figure 17. The results of applying our extended scheme with four authentication bits

Table 5. Evaluation of stego-image quality in worst case by using (2, *n*), (3, *n*) and (4, *n*)-threshold schemes

	(k, n)-threshold schemes	Worst Mean-Square-Error (MSE_{worst})	Worst Peak Signal-to-Noise Ratio ($PSNR_{worst}$)
Eslami and Ahmadabadi's Scheme	(2, *n*) -threshold schemes	$\frac{\sum_{x=1}^{512}\sum_{y=1}^{64}\left[1^2+1^2+1^2+1^2+1^2+1^2+(2^2-1)^2+(2^2-1)^2\right]}{512\times512}=3.00$	$10\times\log_{10}\frac{255^2}{3.00}=43.36$ dB
	(3, *n*) -threshold schemes	$\frac{\sum_{x=1}^{512}\sum_{y=1}^{46}\left[1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+0\right]}{512\times506}=0.91$	$10\times\log_{10}\frac{255^2}{0.91}=48.54$ dB
	(4, *n*) -threshold schemes	$\frac{\sum_{x=1}^{512}\sum_{y=1}^{32}\left[1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2\right]}{512\times512}=0.625$	$10\times\log_{10}\frac{255^2}{0.625}=50.17$ dB
Our Extended Scheme	(2, *n*) -threshold schemes	$\frac{\sum_{x=1}^{256}\sum_{y=1}^{128}\left[1^2+1^2+1^2+1^2+2^2+2^2+2^2+2^2\right]}{512\times512}=2.50$	$10\times\log_{10}\frac{255^2}{2.50}=44.15$ dB
	(3, *n*) -threshold schemes	$\frac{\sum_{x=1}^{256}\sum_{y=1}^{85}\left[1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2\right]}{512\times510}=1.00$	$10\times\log_{10}\frac{255^2}{1.00}=48.13$ dB
	(4, *n*) -threshold schemes	$\frac{\sum_{x=1}^{256}\sum_{y=1}^{64}\left[1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2\right]}{512\times512}=0.75$	$10\times\log_{10}\frac{255^2}{0.75}=49.38$ dB

Acknowledgments

The authors are appreciated to the anonymous reviewers, for their valuable comments and suggestions. This work is partially supported by National Science Council under the grants NSC 98-2221-E-005-050-MY3.

References

[1] M. Naor, and A. Shamir, "Visual cryptography," *Lecture Notes in Computer Science on Advances in Cryptology-EUROCRYPT'94*, vol. 950, pp. 1-12, 1994.

[2] A. Shamir, "How to share a secret," *Communications of the Association for Computing Machinery*, vol. 22, no. 11, pp. 612-613, 1979.

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

[4] C.-C. Thien, and J.-C. Lin, "An image-sharing method with user-friendly shadow images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 12, pp. 1161-1169, 2003.

[5] C.-C. Lin, and W.-H. Tsai, "Secret image sharing with steganography and authentication," *Journal of Systems and Software*, vol. 73, no. 3, pp. 405-414, 2004.

[6] C.-N. Yang, T.-S. Chen, K.-H. Yu, and C.-C. Wang, "Improvements of image sharing with steganography and authentication," *Journal of Systems and Software*, vol. 80, no. 7, pp. 1070-1076, 2007.

[7] C.-C. Chang, Y.-P. Hsieh, and C.-H. Lin, "Sharing secrets in stego images with authentication," *Pattern Recognition*, vol. 41, no. 10, pp. 3130-3137, 2008.

[8] C.-C. Thien, and J.-C. Lin, "A simple and high-hiding capacity method for hiding digit-by-digit data in images based on modulus function," *Pattern Recognition*, vol. 36, no. 12, pp. 2875-2881, 2003.

[9] N.-I. Wu, and M.-S. Hwang, "Data hiding: current status and key issues," *International Journal of Network Security*, vol. 4, no. 1, pp. 1-9, 2007.

[10] C.-C. Wu, S.-J. Kao, W.-C. Kuo, and M.-S. Hwang, "Enhance the Image Sharing with Steganography and Authentication." pp. 1177-1181.

[11] C.-K. Chan, and L. M. Cheng, "Hiding data in images by simple LSB substitution," *Pattern Recognition*, vol. 37, no. 3, pp. 469-474, 2004.

[12] C.-M. Wang, N.-I. Wu, C.-S. Tsai, and M.-S. Hwang, "A high quality steganographic method with pixel-value differencing and modulus function," *Journal of Systems and Software*, vol. 81, no. 1, pp. 150-158, 2008.

[13] C.-F. Lee, and H.-L. Chen, "A novel data hiding scheme based on modulus function," *Journal of Systems and Software*, vol. 83, no. 5, pp. 832-843, 2010.

[14] J.-B. Feng, I.-C. Lin, C.-S. Tsai, and Y.-P. Chu, "Reversible watermarking: current status and key issues," *International Journal of Network Security*, vol. 2, no. 3, pp. 161-170, 2006.

[15] W.-B. Lee, and T.-H. Chen, "A public verifiable copy protection technique for still images," *Journal of Systems and Software*, vol. 62, no. 3, pp. 195-204, 2002.

[16] M. Ulutas, G. Ulutas, and V. V. Nabiyev, “Medical image security and EPR hiding using Shamir's secret sharing scheme,” *Journal of Systems and Software,* vol. 84, no. 3, pp. 341-353, 2011.

[17] C.-C. Chang, K.-N. Chen, C.-F. Lee, and L.-J. Liu, “A secure fragile watermarking scheme based on chaos-and-hamming code,” *Journal of Systems and Software,* vol. In Press, Corrected Proof.

[18] D. Kundur, and D. Hatzinakos, “Digital watermarking for telltale tamper proofing and authentication,” *Proceedings of the IEEE Special Issue on Identification and Protection of Multimedia Information,* vol. 87, no. 7, pp. 1167-1180, July, 1999.

[19] H.-T. Lu, R.-M. Shen, and F.-L. Chung, “Fragile watermarking scheme for image authentication,” *Electronics Letters,* vol. 39, no. 12, pp. 898-900, Jun, 2003.

[20] C.-C. Wu, S.-J. Kao, W.-C. Kuo, and M.-S. Hwang, "A Robust-Fragile Watermarking Scheme for Image Authentication." pp. 176-176.

[21] K.-C. Liao, W.-B. Lee, and C.-W. Liao, “Security of fragile watermarking scheme for image authentication,” *Imaging Science Journal,* vol. 54, no. 3, pp. 129-133, Sep, 2006.

[22] Z. Xinpeng, W. Shuozhong, Q. Zhenxing, and F. Guorui, “Reference Sharing Mechanism for Watermark Self-Embedding,” *IEEE Transactions on Image Processing,* vol. 20, no. 2, pp. 485-495, 2011.

[23] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 5th ed.: Prentice Hall, 2010.

[24] C.-P. Huang, C.-H. Hsieh, and P. S. Huang, “Progressive sharing for a secret image,” *Journal of Systems and Software,* vol. 83, no. 3, pp. 517-527, 2010.

[25] R.-Z. Wang, and C.-H. Su, “Secret image sharing with smaller shadow images,” *Pattern Recognition Letters,* vol. 27, no. 6, pp. 551-555, 2006.

[26] C.-N. Yang, and C.-B. Ciou, “A comment on "Sharing secrets in stegoimages with authentication",” *Pattern Recognition,* vol. 42, no. 7, pp. 1615-1619, 2009.

[27] A. Castiglione, A. De Santis, and C. Soriente, “Security and privacy issues in the Portable Document Format,” *Journal of Systems and Software,* vol. 83, no. 10, pp. 1813-1822, 2010.

[28] A. Castiglione, A. De Santis, and C. Soriente, “Taking advantages of a disadvantage: Digital forensics and steganography using document metadata,” *Journal of Systems and Software,* vol. 80, no. 5, pp. 750-764, 2007.

[29] L. Harn, “Comment on "Multistage secret sharing based on one-way function",” *Electronics Letters,* vol. 31, no. 4, pp. 262, 1995.

[30] A. T. Sherman, and D. A. McGrew, “Key establishment in large dynamic groups using one-way function trees,” *IEEE Transactions on Software Engineering,* vol. 29, no. 5, pp. 444-458, 2003.

[31] J. He, and E. Dawson, “Multisecret-sharing scheme based on one-way function,” *Electronics Letters,* vol. 31, no. 2, pp. 93-95, 1995.

[32] T.-Y. Chang, M.-S. Hwang, and W.-P. Wang, “A New Multi-stage Secret Sharing Scheme Using One-Way Function,” *ACM SIGOPS Operating systems Review,* vol. 39, no. 1, pp. 48-55, 2005.

[33] M.-S. Hwang, C.-C. Chang, and K.-F. Hwang, “A watermarking technique based on one-way hash functions,” *IEEE Transactions on Consumer Electronics,* vol. 45, no. 2, pp. 286-294, 1999.

[34] A. G. Weber. "The USC-SIPI Image Database. Version 5," http://sipi.usc.edu/database/.

[35] Z. Eslami, and J. Z. Ahmadabadi, “Secret image sharing with authentication-chaining and dynamic embedding,” *Journal of Systems and Software*, vol. 84, no. 5, pp. 803-809, 2011.