The Journal of Systems and Software 84 (2011) 803–809

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

Secret image sharing with authentication-chaining and dynamic embedding

Z. Eslami*, J. Zarepour Ahmadabadi

Department of Computer Science, Shahid Beheshti University, G.C., Tehran, Iran

ARTICLE INFO

Article history:
Received 3 March 2010
Received in revised form
14 November 2010
Accepted 1 January 2011
Available online 6 January 2011

Keywords:
Secret image sharing
Steganography
Authentication
Visual quality

ABSTRACT

A popular technique to share a secret image among n participants is to divide it first into some shadow images and then embed the shadows in n"cover" images. The resulting "stego" images, which contain the embedded data, are distributed among intended recipients. In order not to attract any attacker's attention, it is important to apply a suitable embedding such that high quality stego images are produced. Moreover, to ensure the integrity of stego data, a robust authentication mechanism which can detect tampering with high probability should be implemented.

Recently, a series of papers (Lin and Tsai, 2004; Yang et al., 2007; Chang et al., 2008; Yang and Ciou, 2009) have considered polynomial-based secret image sharing with steganography and authentication. The embedding technique employed in all these papers is static, i.e. hidden bits are embedded in predetermined fixed-size blocks of each cover image. It is therefore possible that all the hidden data is replaced in only a subset of blocks of cover images while other blocks remain intact. As for authentication, the best of these schemes detects a tampered stego block with probability 15/16, however, since this is obtained at the cost of using 4 authentication bits per block, the visual quality of stego images is seriously degraded. In this paper, we propose a novel polynomial-based secret image sharing scheme with two achievements. First, a new embedding is proposed so that the block size is determined dynamically according to the size of hidden data and therefore, all the capacity of cover images is used for data hiding. Second, we introduce a new authentication-chaining method which achieves 15/16 as its tamper-detection ability while using only 2 authentication bits. Experimental results are provided to confirm the theory.

1. Introduction

Recent technological advances in computer networks have turned transmission of digital data quite a popular task and digital images are no exception. In particular, there are important confidential images that should be transferred securely over open channels such as the Internet. A common approach to accomplish this is to share the secret image among n entities by using the so-called (t, n)-threshold secret sharing schemes, introduced first independently by Shamir (1979) and Blakley (1979). The secret image is divided (by an entity called the dealer) into n shadow images in such a way that the original secret image can be restored from any t (or more) of shadows, however, the information obtained from $(t-1)$ or fewer shadows is insufficient for reconstruction. This approach adds a fault-tolerant property to image sharing procedure as well. In other words, even if $n-t$ entities are inaccessible for some reasons, the remaining t users are able to restore the secret. In this respect, there are two major approaches. One is visual cryptography schemes in which any t participants may photocopy their shadows on transparencies and stack them on an overhead projector to visually decode the secret image through the human visual system. Naor and Shamir first proposed this idea (Naor and Shamir, 1994). The interested reader can find more on visual cryptography in Yang (2004); Lin and Tsai (2003); Yang and Chen (2006, 2007, 2008); Liu et al. (2010); Tuyls et al. (2005); Tsai et al. (2007); Feng et al. (2008). The other approach is polynomial-based schemes which recover the secret image by adopting Shamir's secret sharing scheme.

However, two problems still exist. First, the above procedure usually produces noise-like shadows which if transmitted would attract attention of attackers. Therefore, for sensitive data some steganographic (data-hiding) method should be utilized to transform shadows to high quality images. In these methods n ordinary images are selected as cover and then the shadow data is embedded in these cover images. The resulting images, called stego images, are then distributed among participants. It is clearly important that high embedding capacity is achieved in a manner that the visual quality of stego images would not be damaged (Lou et al., 2010; Wu and Hwang, 2007). The second problem pertains to ensuring the integrity of stego images during the reconstruction phase. In many applications, any tampering of the stego images should be

* Corresponding author.
E-mail addresses: z_eslami@sbu.ac.ir (Z. Eslami), J.zarepour@mail.sbu.ac.ir (J.Zarepour Ahmadabadi).

0164-1212/$ – see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2011.01.002

Nomenclature

SI	the secret image
t	threshold value, such that t or more shadows can recover SI, while $t-1$ shadows cannot
$P_1, \ldots, P_n$	the participants
CI_i	the cover image corresponding to P_i
$\langle\langle bitstring\rangle\rangle_i$	the leftmost i bits of $bitstring$
$\langle bitstring\rangle_i$	this operator divides $bitstring$ from right to left into substrings of length i and then XORs them to obtain a string of length i, with padding done if necessary
H_K	a collision-free keyed hash function

detected with high probability and this makes a robust authentication technique quite indispensable.

So far, two most popular steganographic embedding methods are the least significant bits (LSB) replacement (Chan and Cheng, 2004; Chang et al., 2003; Wang et al., 2001) and the modulus operation (Wu et al., 2004; Chang et al., 2006; Thien and Lin, 2003). In this paper, we are only concerned with LSB-based methods and provide a literature review of the research done in this category. Thien and Lin (2002) shared the secret image into noise-like shadows using a (t, n)-threshold scheme based on Shamir sharing scheme. In 2004, Lin and Tsai (2004) proposed an image sharing scheme equipped with steganography and authentication. However, their scheme could introduce distortion to the original secret and their authentication ability was rather weak. Afterwards, Yang et al. (2007) proposed an improvement to overcome these defects and enhanced authentication to some degree. In 2008, Chang et al. (2008) employed the Chinese remainder theorem (CRT) to compute authentication bits and obtained better tamper-detection capabilities so that the probability of successful verification for a fake stego block was 1/16. They also claimed to obtain better visual quality for stego images, however, in Yang and Ciou (2009), the authors showed that because of using 4 authentication bits, this quality is indeed degraded. For the sake of completeness, we also mention that in Eslami et al. (2010), Eslami et al. employed the concept of cellular automata to propose an image sharing scheme. They use configurations of the automata to store authentication data and detect a fake stego block with probability 255/256. Therefore, the visual stego quality is enhanced, however, the cost is that consecutive shares must be presented to recover the original secret. In this paper, we only consider polynomial-based schemes.

In all of the above-mentioned schemes, the embedding technique is static, i.e. the cover image is divided into predetermined fixed-size blocks and then the hidden data is embedded into the LSBs of each block. Consequently, it is possible that only a subset of a cover image is used for this purpose and therefore, the number of bits that should be replaced in each block increases unnecessarily. This in turn might have a downside on the visual quality of the resulting stego images. In this paper, we propose a novel embedding method which is dynamic and uses all the capacity of cover images for the purpose of data hiding. Therefore, the block size is determined dynamically according to the size of hidden data and embedding takes place throughout the entire cover image.

We also propose a new authentication method in which chaining of embedded data is used such that computing authentication bits for one block of hidden data depends on previous authentication bits as well. Therefore, while the current best tamper-detection probability (15/16 by Chang et al.) is achieved by allocating 4 authentication bits, our scheme uses only 2 authentication bits to obtain the same result.

X $x = (x_1, x_2 \ldots, x_8)$	V $v = (v_1, v_2 \ldots, v_8)$
W $w = (w_1, w_2 \ldots, w_8)$	Z $x = (x_1, x_2 \ldots, x_8)$

Fig. 1. A 4-pixel block (B) of a cover image.

The rest of the paper is organized as follows: Section 2 reviews briefly recent polynomial-based schemes with steganography and authentication. In Section 3, we explain dynamic embedding, authentication chaining and the proposed secret image sharing scheme. Experimental results are provided in Section 4. Finally, conclusions of the paper are presented in Section 5.

2. Related works

In this section, we review briefly recent secret image sharing schemes. Since all the schemes considered in this section are based on Shamir's sharing scheme, we deliberately omit some detail so as to highlight the essential improvements achieved by our proposed scheme regarding embedding and authentication. We use the following notations throughout the paper.

2.1. Lin et al.'s secret image sharing scheme

This Shamir-based (t, n)-threshold secret image sharing scheme is proposed in 2004 (Lin and Tsai, 2004). Different phases of the scheme are as follows.

2.1.1. Share generation

The secret image SI and cover images.
Each pixel of SI is considered as a secret and is shared by Shamir's (t, n)-scheme Shamir (1979) among $P_1, \ldots, P_n$. Input for Shamir polynomials are pixels of cover images and all calculations are done mod 251.
n shared pixels of the form $(s_1, \ldots, s_8)$ corresponding to each secret pixel.

2.1.2. Computing authentication data for shares

The share $(s_1, \ldots, s_8)$.
The parity bit: $p = s_1 \ldots s_8$.

2.1.3. Embedding of secret data in a given cover image

The cover image (CI) and the secret data (the shares plus corresponding authentication bits).
CI is divided into non-overlapping 4-pixel blocks B as in Fig. 1. Each share $(s_1, \ldots, s_8)$ plus the authentication bit p are embedded in B such that the resulting stego block $\widehat{B}$ in Fig. 2 is obtained. Note that the number of modified bits in the 4 pixels of $\hat{B}$ is (0, 3, 3, 3).
The corresponding stego image.

$\hat{X}$ $\hat{x} = (x_1, x_2 \ldots, x_8)$	$\hat{V}$ $\hat{v} = (v_1, v_2 \ldots, v_5, \boxed{s_1, s_2, s_3})$
$\hat{W}$ $\hat{w} = (w_1, w_2 \ldots, w_5, \boxed{s_4, s_5, s_6})$	$\hat{Z}$ $\hat{z} = (z_1, z_2 \ldots, z_5, \boxed{s_7, s_8, p})$

Fig. 2. Stego block ($\hat{B}$) of Lin et al.'s scheme.

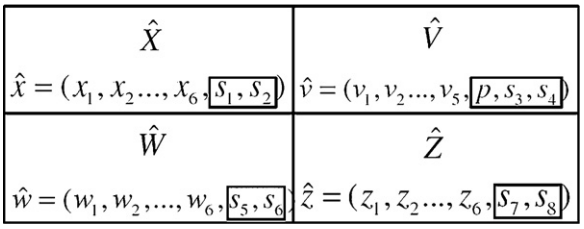

Fig. 3. Stego block ($\hat{B}$) of Yang et al.'s scheme.

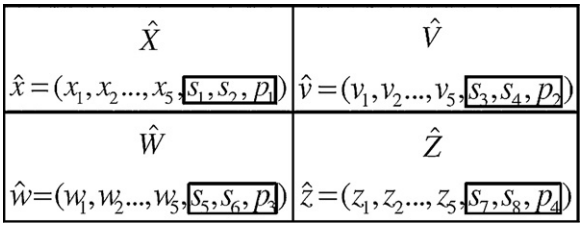

Fig. 4. Stego block ($\hat{B}$) of Chang et al.'s scheme.

2.2. Yang et al.'s image sharing scheme

Proposed in 2007 Yang et al. (2007), the share generation is the same as Lin et al. with a slight difference in Shamir polynomials input. One authentication bit is generated using a keyed hash function, resulting in 1/2 as possibility of successful tampering of each stego block. The embedding strategy is essentially the same as Lin et al. with each stego block as in Fig. 3. The number of modified bits in the 4 pixels of $\hat{B}$ is (2, 3, 2, 2).

2.3. Chang et al.'s image sharing scheme

The scheme is presented by Chang et al. (2008). The share generation is the same as the previous methods, except that the shares are generated from a group of t consecutive pixels of SI (and another slight difference in Shamir's polynomial's input). Here, 4 authentication bits are computed using the Chinese remainder theorem (CRT). This reduces possibility of successful tampering of each stego block to 1/16, however, due to increase in the number of authentication bits, the visual quality is affected. The embedding strategy is essentially the same as Lin et al. with each stego block as in Fig. 4. The number of modified bits in the 4 pixels of $\hat{B}$ is (3, 3, 3, 3).

3. The proposed scheme

In this section, we explain in detail dynamic embedding, authentication chaining and the proposed secret image sharing scheme.

3.1. Dynamic embedding

Suppose that we want to hide $D_1, \ldots, D_l$ into a given cover image CI, where each D_i consists of one byte $(s_1, \ldots, s_8)$ plus m-bit authentication string $(a_1, \ldots, a_m)$, where m is usually between 1 to 3. We require that the embedding be done such that first, the visual quality of CI is not degraded and second, all the capacity of CI is used for hiding. Since the embedding techniques of Section 2 cannot accomplish these goals, we propose the following method.

In dynamic embedding, as in all the methods in Section 2, we consider l blocks in CI and then embed each D_i into least significant bits (LSBs) of pixels of these blocks. However, unlike previous schemes, we do not consider 4 as the size of each block. We determine the block size dynamically so that our goals are met. In order to use all the capacity of CI, we define the size of each block of CI as

$$BS = \left\lfloor \frac{|CI|}{l} \right\rfloor,$$

where $|CI|$ represents the number of pixels in CI. Now, each D_i is to be embedded in a block of CI with BS pixels. Let $B = (B_1, \ldots, B_{BS})$ represent a block where each $B_{i,1 \le i \le BS}$ is a pixel. We next determine the number of LSB bits that are replaced with hidden data in each B_i. Since we have $|D_i|$ (which is $8 + m$) bits to hide in BS pixels of B, we define Nb as

$$Nb = \frac{|D_i|}{BS}.$$

Clearly, in each pixel of a block, at least $\lfloor Nb \rfloor$ LSB bits are replaced with D_i. However, if Nb is not an integer, then in $|D_i| - BS.\lfloor Nb \rfloor$ pixels, one more bit must be used to embed the remaining bits of D_i, i.e. in these pixels $\lceil Nb \rceil$ bits will be used for embedding. Therefore, we define the number of used bits in a pixel B_i of B as Ub_i where

$$Ub_i = \begin{cases} \lfloor Nb \rfloor, & \text{if } i = 1, \ldots, BS * \lceil Nb \rceil - |D_i|, \\ \lceil Nb \rceil, & \text{otherwise.} \end{cases}$$

Note that for convenience we use $\lceil Nb \rceil$ bits in the last pixels of B. As an example, suppose that there are $l = 2^{15}$ 10-bits data (8-bits shares plus 2-bits authentication) that are to be embedded in a 512×512 cover image. Then $BS = 8$, $|D_i| = 10$, $|CI| = 2^{18}$ and $Nb = 1.25$. Therefore, CI is divided into blocks of 8 pixels each. Now, 10 bits must be embedded in each 8-pixel block. Hence, in 6 pixels only one LSB will be used while in the remaining 2 pixels, 2 LSB bits are replaced, i.e. the number of modified bits in an 8-pixel block will be (1, 1, 1, 1, 1, 1, 2, 2) (Fig. 5). Now, compare this result with schemes of Section 2. Their block size is 4, so for hiding 10 bits in a 4-pixel block, 2 pixels will have 3 LSBs replaced, i.e. the number of modifies bits in a 4-pixel block should be (2, 2, 3, 3). This means that a better visual quality is guaranteed by dynamic embedding.

Another advantage of dynamic embedding is that we no longer have any restriction on the size of cover images, while in static embedding, it should hold that $|CI| = 4l$. Therefore, in cases where block size is less than 4 and therefore static embedding cannot be used, dynamic embedding is still applicable.

3.2. Authentication chaining

In order to ensure that a given stego image is not tampered, we should hide some authentication data in its blocks. The larger the size of authentication data, the greater the tamper detection abilities of the scheme. However, embedding of more bits downgrades the visual quality of the image.

The authentication techniques of the methods in Section 2 is such with a authentication bits in each block of cover image, the probability of successful tampering is 2^{-a}. Our goal is to use chaining of authentication bits so that with a authentication bits, we achieve 2^{-2a} as the probability of successful tampering. We

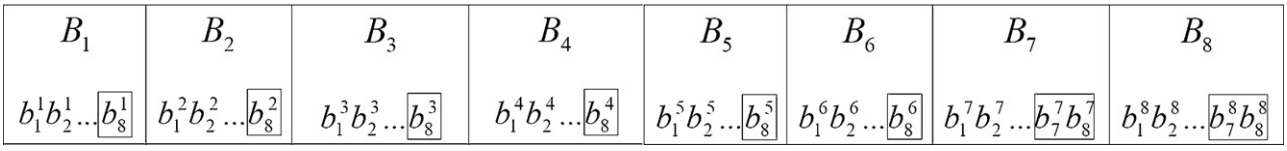

Fig. 5. Number of tampered bits in each pixel.

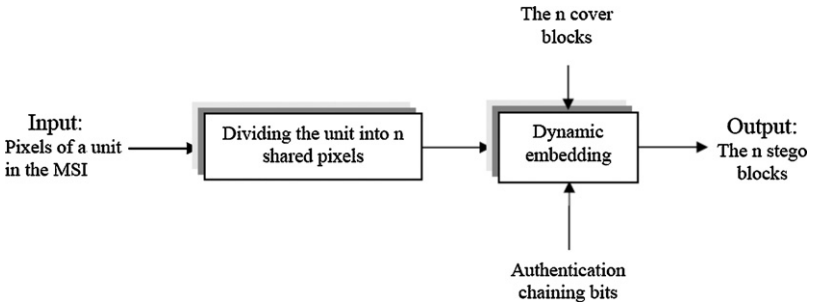

Fig. 6. Diagram of the sharing and embedding algorithm.

generate 2 authentication bits, a_1a_2, for a stego block $\hat{B}$ corresponding to a cover image block B. These bits are computed such that tampering of $\hat{B}$'s pixels affects authentication bits of two blocks, namely those of $\hat{B}$ and the next block. Therefore, the probability of successful tampering at this stage is about 1/16.

Let $B^j = (B^j_1, \ldots, B^j_{BS})$ be the jth block of CI_i. Suppose that SH^j_i is the share of P_i to be embedded in B^j. Then, the authentication bits for B^j is denoted by Aut^j_i and are computed as follows:

$$Aut^j_i = \left\langle H_k(SH^{j-1}_i || SH^j_i || \langle\langle B^j_1 \rangle\rangle_{M_1} || \langle\langle B^j_2 \rangle\rangle_{M_2} || \cdots || \langle\langle B^j_{BS} \rangle\rangle_{M_{BS}} || j) \right\rangle_2 \quad (1)$$

where $M_{k_{1\le k\le BS}} = 8 - Ub_k$ is the number of intact bits of B^j_k. We also define SH^0_i to be all-zero vector. Note that the key K for the keyed hash function H_K must be shared using a (t, n)-sharing scheme as well. Since Aut^j_i corresponding to the jth block of a given stego image STG_i can be computed exactly the same way, we do not require to have any assumptions about the collision properties of the hash function.

3.3. The proposed secret image sharing scheme

In this section, we combine a polynomial-based (t, n)-secret sharing scheme with dynamic embedding and authentication chaining to propose a new secret image sharing scheme with improved performance. The scheme consists of two phases: (1) sharing and embedding phase and (2) Verification and recovery phase.

3.3.1. Sharing and embedding phase

The shares can be generated the same as Chang et al., i.e. the secret image (SI) is processed so that a modified secret image (MSI) is produced in which all pixel values are in range 0–250, then MSI is shared and embedded together with additional authentication data into cover images as in Fig. 6, however embedding and authentication are done as in Sections 3.1 and 3.2. The details are as follows.

MSI is divided into t-pixel units U. Then, for each unit U, a polynomial of degree $(t-1)$ such as $h(x) = \sum_{i=0}^{t-1} h_i x^i \pmod{251}$ is constructed where the coefficients h_i are the pixels of U. Now, we must choose n distinct input x_i so that $(h(x_1), \ldots, h(x_n)) = (SH^U_1, SH^U_2, \ldots, SH^U_n)$ are produced as shares and given to n participants $P_1, \ldots, P_n$. The share of P_k corresponding to U must be embedded into a block of CI_k such as B^U. In our scheme, we take the intact bits of the first pixel of B^U as x_k.

Suppose that $l = \lceil |MSI|/t \rceil$. Therefore, we have in total l units $U^1, \ldots, U^l$. For each unit, n shares are produced, so $SH^{U^i}_1, SH^{U^i}_2, \ldots, SH^{U^i}_n$, $1 \le i \le l$ are all the $n \times l$ shares. Hence, for the kth participant P_k, we have to embed $SH^{U^1}_k, SH^{U^2}_k, \ldots, SH^{U^l}_k$ in blocks of his corresponding cover image CI_k. We determine the block size (BS), and the number of used bits in each pixel of a block as explained in Section 3.1. Then CI_k is divided into blocks of BS pixels each and $SH^{U^j}_k$ is embedded into the jth block of CI_k, say B^j_k. We also generate a 2-bits authentication string Aut^j_k corresponding to B^j_k using the method of Section 3.2. Therefore, in total, 10 bits $(SH^{U^j}_k \cup Aut^j_k)$ are to be hidden in the block B^j_k such that in each pixel $\lfloor Nb \rfloor$ (or $\lceil Nb \rceil$) LSBs are used for embedding. Note that the block size BS is determined for each CI_k so that cover images can be of different sizes.

3.3.2. Verification and recovery phase.

The details of this phase are depicted in Fig. 7. Without loss of generality, suppose that t stego images $STG_1, \ldots, STG_t$ are presented to recover the original secret image SI. Each STG_i is divided

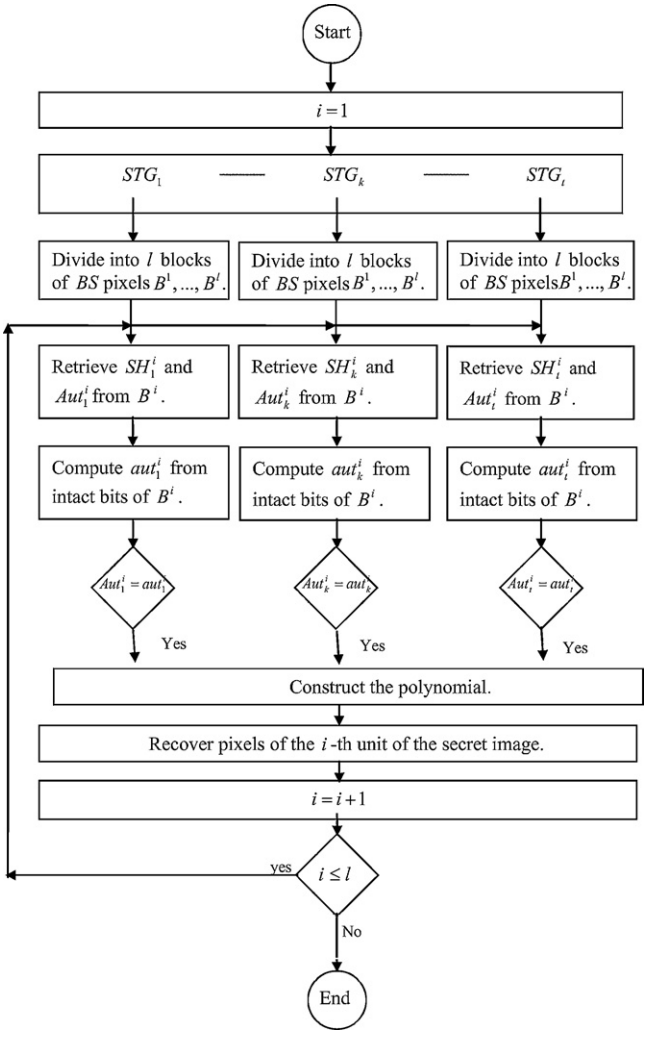

Fig. 7. Diagram of the verification and recovery phase.

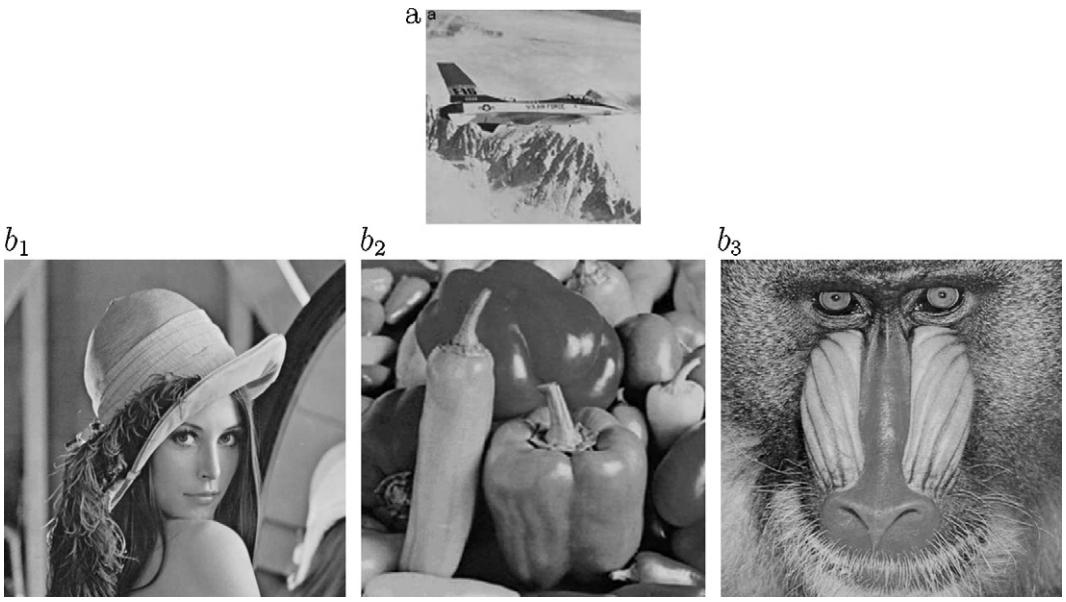

Fig. 8. (a) The secret image (*SI*), 256 × 256; (b) Cover images, Lena, Pepper and Baboon 512 × 512.

into a set of blocks with *BS* pixels. Let $B^j = (B^j_1, \ldots, B^j_{BS})$ be the *j*th block of STG_i. We retrieve from this block the hidden share SH^j_i and Aut^j_i. Then we compute Aut^j_i once again from intact bits of these pixels as described in Section 3.2. We then compare the two authentication values. If they differ, then a tampering has occurred, otherwise the block is successfully verified and the shared pixel (SH^i_j) is retrieved and accepted. When all *t* shares corresponding to the *j*th block of all *t* stego images are obtained, then Lagrange interpolation can be used to recover the *j*th unit of *SI*. We repeat this procedure until all units are restored. Note that since we have basically used Shamir's sharing scheme to generate the shares, less than *t* stego images can derive no information about the secret image.

4. Experimental results

The criterion for the visual quality of the stego images is the peak-signal-to-noise ratio (PSNR) defined as

$$\mathrm{PSNR} = 10 \times \log_{10} \frac{(255)^2}{\mathrm{MSE}} dB, \tag{2}$$

where MSE is the mean-square error between the cover image and the stego image. If the cover image is sized $f * g$, MSE is defined as

$$\mathrm{MSE} = \frac{1}{f \times g} \sum_{i=1}^{f} \sum_{j=1}^{g} (x_{ij} - y_{ij})^2, \tag{3}$$

where x_{ij} and y_{ij} denote the cover and the stego pixel values, respectively.

In Yang and Ciou (2009), all the schemes of Section 2, i.e. Lin et al., Yang et al., and Chang et al. are compared using a (2, 3)-secret sharing scheme, tested by a 256 × 256-pixel secret image Jet (Fig. 8(a)) and three 512 × 512 cover images Lena, pepper and baboon (Fig. 8(b)). The results show that regarding stego visual quality $\mathrm{PSNR}_{\mathrm{Yang}} > \mathrm{PSNR}_{\mathrm{Lin}} > \mathrm{PSNR}_{\mathrm{Chang}}$. In this section, we conduct the same experiment with our proposed scheme. Here, with the same notation as in Section 3.1, we have:

$$l = \left\lceil \frac{|MSI|}{t} \right\rceil = \left\lceil \frac{2^{16}}{2} \right\rceil = 2^{15},$$
$$BS = \left\lfloor \frac{|CI|}{l} \right\rfloor = \left\lceil \frac{2^{18}}{2^{15}} \right\rceil = 8,$$
$$|D_i| = 8 + 2 = 10,$$
$$Nb = \frac{|D_i|}{BS} = \frac{|10|}{8} = 1.25,$$
$$Ub_{i_{1 \le i \le 6}} = 1, \quad Ub_7, \quad Ub_8 = 2.$$

Therefore, each cover image is divided into blocks of 8 pixels each and then 10-bits shares will be embedded in each block. Hence, the number of modified bits in a block will be (1, 1, 1, 1, 1, 1, 2, 2). This is better than existing methods which embed the same 10 bits in 4-pixel blocks. The experimental results show that $\mathrm{PSNR}_{\mathrm{Ours}} > \mathrm{PSNR}_{\mathrm{Yang}} > \mathrm{PSNR}_{\mathrm{Lin}} > \mathrm{PSNR}_{\mathrm{Chang}}$ (Fig. 9). The results are summarized in Table 1.

Table 1
Results for (2,3)-secret sharing schemes.

Scheme	PSNR (db)		
	Lena	Pepper	Baboon
Lin et al.	42.29	42.27	42.28
Yang et al.	44.62	44.58	44.57
Chang et al.	40.52	40.25	40.21
Ours	48.13	48.12	48.10

Table 2
Results for (3,4)-secret sharing schemes.

Scheme	PSNR (db)			
	Lena	Pepper	Baboon	Flower
Lin et al.	44.66	44.63	44.65	44.73
Yang et al.	46.97	46.95	46.93	46.89
Chang et al.	42.70	42.09	42.30	41.68
Ours	51.94	51.93	51.93	51.93

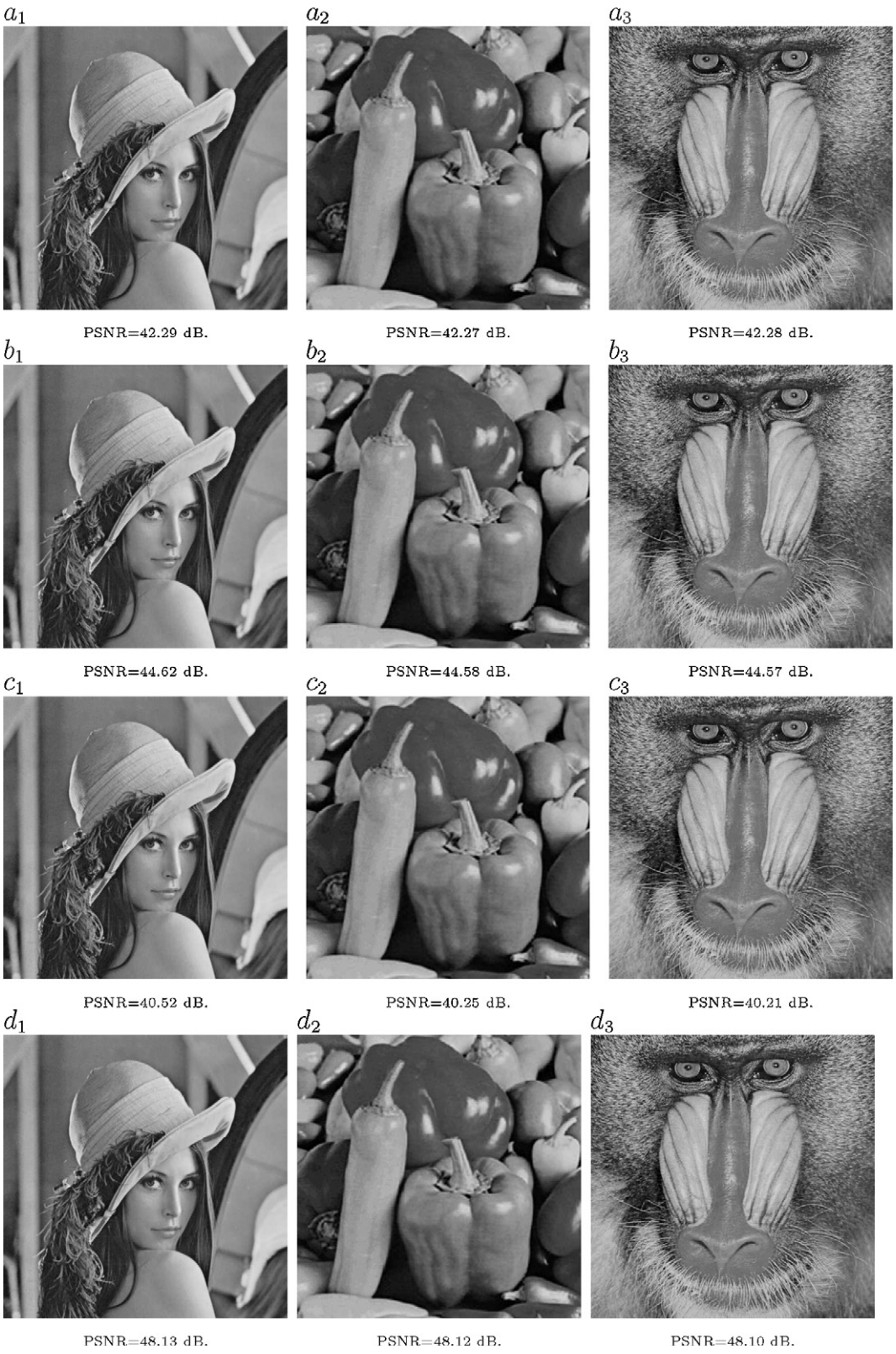

Fig. 9. Stego images for (2, 3) scheme: (a) Lin et al.'s scheme, (b) Yang et al.'s scheme, (c) Chang et al.'s scheme, and (d) Our scheme.

We also test our scheme in a (3, 4)-threshold secret sharing scheme. In Eslami et al. (2010), the scheme of Chang et al. is tested using a (3, 4)-secret sharing scheme (with Flower as the fourth cover image) and the reported PSNR is about 42. We conduct the same experiment in the proposed scheme with l, BS, D_i and Nb as in the following and obtain PSNR as high as 52 (Table 2). The reason for this visual quality is that here dynamic embedding embeds 10-bits data in 11 pixels of each cover image in such a way that at most one LSB in each pixel is replaced by hidden

data.

$$l = \left\lceil \frac{|MSI|}{t} \right\rceil = \left\lceil \frac{2^{16}}{3} \right\rceil = 21,846,$$
$$BS = \left\lfloor \frac{|CI|}{l} \right\rfloor = \left\lfloor \frac{2^{18}}{21,846} \right\rfloor = \lfloor 11.9 \rfloor = 11,$$
$$|D_i| = 10,$$
$$Nb = \frac{|D_i|}{BS} = \frac{|10|}{11} = 0.9,$$
$$Ub_1 = 0, \quad Ub_{i_{2 \le i \le 8}} = 1.$$

In term of verification ability, we employ chaining of authentication bits and achieve 1/16 as in Chang et al.'s scheme which is the best among existing methods.

5. Conclusions

The embedding techniques based on LSB replacement employed so far are static, i.e. hidden bits are embedded in predetermined fixed-size blocks of each cover image. It is therefore possible that all the hidden data is replaced in only a subset of blocks of cover images while other blocks remain intact and this in turn might have a downside on the visual quality of the resulting stego images. In this paper, we propose a novel embedding method in which the block size is determined dynamically according to the size of hidden data so that all the capacity of a cover image is used for the purpose of embedding to guarantee a better visual quality.

The probability of successful tampering in existing schemes which use a authentication bits in each block of cover image is 2^{-a}. In this paper, we propose chaining of authentication bits as a feedback mechanism so that computing authentication bits for one block depends on previous blocks as well. Therefore, tampering in one block affects authentication data of the next block too and this results in reducing successful tampering probability down to 2^{-2a}.

References

Blakley, G., 1979. Safeguarding cryptographic keys. In: AFIPS Conference Proceedings, vol. 48 , pp. 313–317.

Chan, C., Cheng, L., 2004. Hiding data in images by simple lsb substitution. Pattern Recognition 37, 474–496.

Chang, C., Chan, C., Fan, Y., 2006. Image hiding scheme with modulus function and dynamic programming. Pattern Recognition 39, 1155–1167.

Chang, C., Hsiao, J., Chan, C., 2003. Finding optimal least-significant-bits substitution in image hiding by dynamic programming strategy. Pattern Recognition 36, 1583–2159.

Chang, C., Hsieh, Y., Lin, C., 2008. Sharing secrets in stego images with authentication. Pattern Recognition 41, 3130–3137.

Eslami, Z., Razzaghi, S., Ahmadabadi, J.Z., 2010. Secret image sharing based on cellular automata and steganography. Pattern Recognition 43, 397–404.

Feng, J., Wu, H., Tsai, C., Chang, Y., Chu, Y., 2008. Visual secret sharing for multiple secrets. Pattern Recognition 41, 3572–3581.

Lin, C., Tsai, W., 2003. Visual cryptography for gray-level images by dithering techniques. Pattern Recognition Letters 24, 349–358.

Lin, C., Tsai, W., 2004. Secret image sharing with steganography and authentication. The Journal of Systems and Software 73, 405–414.

Liu, F., Wu, C., Lin, X., 2010. Some extensions on threshold visual cryptography schemes. The Computer Journal 53, 107–119.

Lou, D.-C., Wu, N.-I., Wang, C.-M., Lin, Z.-H., Tsai, C.-S., 2010. A novel adaptive steganography based on local complexity and human vision sensitivity. Journal of Systems and Software 83, 1236–1248.

Naor, M., Shamir, A., 1994. Visual cryptography. In: Advances in Cryptology - EUROCRYPT'94 LNCS, vol. 950 , pp. 1–12.

Shamir, A., 1979. How to share a secret. Communication of the ACM 22, 612–613.

Thien, C., Lin, J., 2002. Secret image sharing. Computer Graphics 26 (5), 765–770.

Thien, C., Lin, J., 2003. An image-sharing method with user-friendly shadow images. IEEE Transactions on Circuits and Systems, 1161–1169.

Tsai, D., Chen, T., Horng, G., 2007. A cheating prevention scheme for binary visual cryptography with homogeneous secret images. Pattern Recognition 40, 2356–2366.

Tuyls, P., Hollmann, H., Lint, J.V., Tolhuizen, L., 2005. Xor-based visual cryptography schemes, Designs. Codes and Cryptography 37, 169–186.

Wang, R., Lin, C., Lin, J., 2001. Image hiding by optimal lsb substitution and genetic algorithm. Pattern Recognition 34, 671–683.

Wu, N.-I., Hwang, M.-S., 2007. Data hiding: current status and key issues. International Journal of Network Security 4, 1–9.

Wu, Y., Thien, C., Lin, J., 2004. sharing and hiding secter images with size constraint. Pattern Recognition 37, 1377–1385.

Yang, C., 2004. New visual secret sharing schemes using probabilistic method. Pattern Recognition Letters 25, 481–494.

Yang, C., Chen, T., 2006. Reduce shadow size in aspect ratio invariant visual secret sharing schemes using a square block-wise operation. Pattern Recognition 39, 1300–1314.

Yang, C., Chen, T., 2007. Extended visual secret sharing schemes: improving the shadow image quality. International Journal of Pattern Recognition and Artificial Intelligence 21, 879–898.

Yang, C., Chen, T., 2008. Colored visual cryptography scheme based on additive color mixing. Pattern Recognition 41, 3114–3129.

Yang, C., Chen, T., Yu, K., Wang, C., 2007. Improvements of image sharing with steganography and authentication. The Journal of Systems and Software 80, 1070–1076.

Yang, C.-C., Ciou, C.-B., 2009. A comment on "sharing secrets in stego images with authentication". Pattern Recognition 42, 1615–1619.