Contents lists available at SciVerse ScienceDirect



Optics Communications

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

Steganography and authentication in image sharing without parity bits

Ching-Nung Yang ^{a,*}, Jin-Fwu Ouyang ^a, Lein Harn ^b

^a CSIE Dept., National Dong Hwa University, Taiwan

^b CSEE Dept., University of Missouri, Kansas City, USA

ARTICLE INFO

Article history: Received 27 March 2011 Received in revised form 10 October 2011 Accepted 1 December 2011 Available online 16 December 2011

Keywords: Secret image sharing steganography Authentication Stego-image Parity bit Symmetric bivariate polynomial

1. Introduction

Steganography comes from the Greek word steganos, and means concealed writing. It is an important research subject in the field of cryptography and information security. Steganography hides the secret message into a cover media to generate a stego-media, on which the existence of the embedded secret cannot be detected. Steganographic technique can overcome the conventional cryptographic approach, providing new solutions for secure data transmission without being suspect to censors. The cover media in a steganography scheme could be image, audio, video, document, ..., etc. However, if the stego-media is lost or corrupted, the secret data cannot be reconstructed. Therefore, several secret sharing techniques have been proposed to overcome this weakness. In this paper, we discuss a secret image sharing scheme with steganography and authentication functions. The cover media in our scheme is a digital image (referred to as cover image), and the image which the secret data is embedded is called as a stego-image.

A (k, n) secret image sharing scheme, where $k \le n$, divides a secret message into n shadow images in such a way that one can reconstruct the secret image with any k or more than k shadow images, but one cannot obtain any information of the secret image from fewer than k shadow images. There are two major categories in secret image sharing scheme: one is the visual cryptography, and the other is the polynomial-based

E-mail address: cnyang@mail.ndhu.edu.tw (C.-N. Yang).

ABSTRACT

Recently, a polynomial-based (k, n) steganography and authenticated image sharing (SAIS) scheme was proposed to share a secret image into n stego-images. At the same time, one can reconstruct a secret image with any k or more than k stego-images, but one cannot obtain any information about the secret from fewer than k stego-images. The beauty of a (k, n)-SAIS scheme is that it provides the threshold property (i.e., k is the threshold value), the steganography (i.e., stego-images look like cover images), and authentication (i.e., detection of manipulated stego-images). All existing SAIS schemes require parity bits for authentication. In this paper, we present a novel approach without needing parity bits. In addition, our (k, n)-SAIS scheme provides better visual quality and has higher detection ratio with respect to all previous (k, n)-SAIS schemes.

© 2011 Elsevier B.V. All rights reserved.

secret image sharing. In visual cryptography, participants may photocopy their shadows on transparencies and stack them on an overhead projector to visually decode the secret through the human visual system without hardware and computation. Although visual cryptography has the staking-to-see property, it has the poor visual quality of a reconstructed image. On the contrary, the polynomial-based secret image sharing can recover a distortion-less secret image by using Lagrange interpolation. The polynomial-based secret image sharing is to hide secret pixels as constant terms in (k-1)-degree polynomials using Shamir's secret sharing scheme [1]. The authors in [2] used all coefficients in a (k-1)-degree polynomial for embedding secret pixels, and reduced shadow images to size 1/k of the size of the secret image. In [3], the size of shadow image was further reduced by using Huffman code. Shadow images in [2,3] are noise-like and suspect to censorships. Therefore, it is desirable to design a (k, n) secret image sharing scheme using steganography so that shadow images looks like a cover image. Moreover, if we consider the authentication ability to detect the manipulation of shadow images, this scheme is called a (k, n) steganographic and authenticated image sharing (SAIS) scheme. Such a (k, n)-SAIS scheme has three key properties. First, it has the steganography property. The proposed (k, n)-SAIS scheme has meaningful shadow images that look like cover images. In this paper, we use the term "stego-images" in steganography to represent shadow images in the proposed SAIS scheme. Second, it has the threshold property that one can reconstruct the secret image from any *k* stego-images. However, any (k-1)or fewer than (k-1) stego-images cannot reveal any information about the secret image. Lastly, it has authentication ability. One can verify the correctness of stego-images to prevent any accidentally generating error stego-images or any intentionally presenting fake stego-images.

^{*} Corresponding author at: Department of Computer Science and Information Engineering, National Dong HwaUniversity, #1, Sec. 2, Da Hsueh Rd., Hualien, Taiwan. Tel.: + 886 3 8634025; fax: + 886 3 8634010.

^{0030-4018/\$ -} see front matter © 2011 Elsevier B.V. All rights reserved. doi:10.1016/j.optcom.2011.12.003

Some (k, n)-SAIS schemes were proposed in [4–7]. Lin et al. embedded eight shared bits and one parity bit (for a total of nine bits) in a four-pixel block to construct a (k, n)-SAIS scheme [4]. For precise presentation, in this paper we refer to the parity bits as authentication bits. Embedding nine bits in a four-pixeled block is intended to make the stego-image look like the cover image. On the other hand, one parity bit is used to detect whether this block has been tampered with. Although any accidentally generating wrong stego-images can be detected, the intentionally modifying stego-images by dishonest shareholders can be hard to detect. In [5], Yang et al. solved this dishonesty problem and simultaneously improved the visual quality of stego-images by rearranging nine bits in a block. Afterward, the schemes in [6,7] enhanced authentication ability and the visual quality of stego-images. Chang et al. [6] enhanced the detection ratio of manipulated blocks in a fake stego-image by using four authentication bits in a block based on the Chinese remainder theorem. In [7], the authors employed linear cellular automata, digital signatures, and hash functions to reduce the number of modified bits in a block so that better visual quality can be guaranteed.

All existing (k, n)-SAIS schemes [4–6] require additional authentication bits to authenticate stego-images. In [7], the SAIS scheme enables double authentication. One can verify the signature of a stego-image to assure the integrity of each stego-image by using a public key. However, an invalid signature only shows the stego-image that has been modified; but not the tampered stego-blocks. In this paper, we consider a (k, n)-SAIS scheme without any additional authentication bits. All involved participants perform authentication with a vote-based protocol. Every participant authenticates other shadows, and votes for stegoimages that he/she trusts. According to the result of a majority vote, the authenticity of the stego-images is then determined by peers. The proposed (k, n)-SAIS is based on symmetric bivariate polynomials and it enhances the visual quality of stego-images. The following sections are organized as follows. In Section 2, we review existing (k, n)-SAIS schemes. The motivation and key contributions of our paper are described in Section 3. The proposed (k, n)-SAIS scheme and the concept of bivariate polynomial are introduced in Section 4. Details on the evaluation and experiment are given in Section 5, and Section 6 is the conclusion.

2. Related works

2.1. Shamir's Secret Sharing Scheme

In 1979, Shamir [1] published a landmark paper that a (k, n) secret-sharing scheme can be constructed by hiding a secret data in the constant term f_0 of a (k-1)-degree polynomial $f(x)=(f_0+f_1x+...+f_{k-1}x^{k-1}) \mod P$, where *P* is a prime number. By using $i \in [1, n]$, a dealer can generate *n* shadows as (i, f(i)), for i = 1, 2, ..., n. Any *k* shadows (say (1, f(1)), ..., (k, f(k))) can jointly reconstruct this (k - 1)-degree polynomial f(x) following Lagrange interpolation formula and the secret data can be derived from $f_0 = f(0)$.

$$f(x) = \sum_{i=1}^{k} f(i) \prod_{1 \le j \le k, \ j \ne i} \frac{(x-i)}{(j-i)} \operatorname{mod} P.$$

With this (k-1)-degree univariate polynomial, Thien et al. [2] embedded secret pixels into k coefficients in f(x). Thien et al.'s (k, n) secret image sharing scheme is briefly described below. We first divide a secret image into b non-overlapping k-pixel blocks, and every j-th $(0 \le j \le b - 1)$ block includes the secret pixels $(s_{jk}, s_{jk+1}, ..., s_{jk+k-1})$. The (k-1)-degree polynomial $f_j(x) = (s_{jk}+s_{jk+1}x+s_{jk+2}x^2+...+s_{jk+k-1}x^{k-1})GF(2^8)$ represents a shadow pixel associated with this j-th block, where x is often a shadow identity. By choosing n shadow identities $i \in [1, n]$, we then obtain n shadow pixels $f_j(i)$. We repeat this process for all b blocks

to generate *n* shadows. Obviously, shadow size is 1/k of the size of the secret image since we embed *k* secret pixels to one shadow pixel each time. In decryption, the polynomial $f_j(x)$ can be reconstructed from any *k* shadow pixels so that we can recover the secret image. Here, we use the Galois Field $GF(2^8)$ to embed 256 grayscales in a secret image without distortion. Some schemes adopt an ordinary arithmetic operation (i.e., mod 251) for simple calculation. However, under mod 251, the gray-scale values > 250 should be truncated to 250 and this causes distortion. In this paper, we use $GF(2^8)$ for our experiment to derive a loss-less secret image.

2.2. Lin et al.'s (k, n)-SAIS Scheme

Lin et al. proposed a (k, n)-SAIS scheme [4] by combining steganography and authentication to prevent fake stego-images. Every ksecret pixel is embedded into $(f_0, f_1, ..., f_{k-1})$ in a (k-1)-degree Shamir's polynomial f(x). The output of f(x) is an eight-bit tuple $(s_1, s_2, ..., s_8)$, and one parity bit p for authentication which is even or odd parity depending on the binary parity sequence generated by the secret key. To reduce the distortion of image as much as possible, nine bits (i.e., eight shared bits and one parity bit) are embedded into a fourpixeled stego-block. The input of the polynomial is chosen from the upper left pixel in this stego-block, and nine bits replace the last three least significant bits (LSBs) in the other three pixels to construct a stego-image.

Suppose that the original four-pixeled block in a cover image is $\frac{\overline{X|V|}}{W|Z|}$, where $X=(x_1, x_2, \dots, x_8)$, $V=(v_1, v_2, \dots, v_8)$, $W=(w_1, w_2, \dots, w_8)$, and $Z=(z_1, z_2, \dots, z_8)$. Then, insert (s_1, s_2, \dots, s_8) and p into $\frac{\overline{X|V|}}{W|Z|}$ to derive $\hat{X}, \hat{V}, \hat{W}$, and \hat{Z} as follows.

$$\begin{cases} \hat{X} = (\hat{x}_1, \hat{x}_2, \cdots, \hat{x}_8) = (x_1, x_2, \cdots, x_8), \\ \hat{V} = (\hat{v}_1, \hat{v}_2, \cdots, \hat{v}_8) = (v_1, v_2, \cdots, v_5, p, s_1, s_2), \\ \hat{W} = (\hat{w}_1, \hat{w}_2, \cdots, \hat{w}_8) = (w_1, w_2, \cdots, w_5, s_3, s_4, s_5), \\ \hat{Z} = (\hat{z}_1, \hat{z}_2, \cdots, \hat{z}_8) = (z_1, z_2, \cdots, z_5, s_6, s_7, s_8). \end{cases}$$
(1)

The stego-block block $\frac{|\hat{x}|\hat{v}|}{|\hat{y}|\hat{z}|}$ of the Lin et al.'s (k, n)-SAIS scheme is shown in Fig. 1, where the underlining of $\hat{x} = (\underline{x_1, x_2, \cdots, x_8})$ implies that (x_1, x_2, \cdots, x_8) is the input of f(x).

2.3. Yang et al.'s (k, n)-SAIS Scheme

Because a participant can derive parity information from his/her own stego-image, he/she can maliciously make a fake stego-image that can pass authentication but compromises the reconstruction. In [5], Yang et al. avoided this authentication weakness to prevent dishonest participants from cheating. The enhanced authentication ability comes from the use of a hash function with a secret key to generate the authentication bit p. The inputs of HMAC are the block ID and all 31 bits in the stego-block exclusive to the authentication bit. Yang et al.'s (k, n)-SAIS scheme also rearranged nine bits in the stego-block to improve the visual quality of the stego-image.



Fig. 1. The stego-block of Lin et al.'s (k, n)-SAIS scheme.



Fig. 2. The stego-block of Yang et al.'s (*k*, *n*)-SAIS scheme.

The modified pixels \hat{X} , \hat{V} , \hat{W} , and \hat{Z} in a stego-block are shown in Eq. (2).

$$\begin{cases} \hat{X} = (\hat{x}_1, \hat{x}_2, \dots, \hat{x}_8) = (x_1, x_2, \dots, x_6, s_1, s_2), \\ \hat{V} = (\hat{v}_1, \hat{v}_2, \dots, \hat{v}_8) = (v_1, v_2, \dots, v_5, p, s_3, s_4), \\ \hat{W} = (\hat{w}_1, \hat{w}_2, \dots, \hat{w}_8) = (w_1, w_2, \dots, w_5, w_6, s_5, s_6), \\ \hat{Z} = (\hat{z}_1, \hat{z}_2, \dots, \hat{z}_8) = (z_1, z_2, \dots, z_6, s_7, s_8). \end{cases}$$

$$(2)$$

The stego-block is shown in Fig. 2. Note that $(\underline{x_1, ..., x_6})$ in X and $(\underline{w_5, w_6})$ in W are used as the eight-bit input for f(x). The number of modified bits in \hat{X} , \hat{V} , \hat{W} and \hat{Z} are (2, 3, 2, 2) which is different from (0, 3, 3, 3) in Lin et al.'s scheme. This uniform arrangement of nine pixels reduces distortion, and thus Yang et al.'s scheme enhance the visual quality of the stego-image.

2.4. Chang et al.'s (k, n)-SAIS Scheme

To enhance authentication ability, Chang et al.'s scheme [6] used four authentication bits in a stego-block. However, the schemes in [4,5] only use one authentication bit in a stego-block. In [6], four CRT-based authentication bits are computed and combined with watermark bits to produce four parity bits (p_1, p_2, p_3, p_4). Modified pixels $\hat{X}, \hat{V}, \hat{W}$, and \hat{Z} are shown in Eq. (3).

$$\begin{cases} \hat{X} = (\hat{x}_1, \hat{x}_2, \dots, \hat{x}_8) = (x_1, x_2, \dots, x_5, s_1, s_2, p_1), \\ \hat{V} = (\hat{v}_1, \hat{v}_2, \dots, \hat{v}_8) = (v_1, v_2, \dots, v_5, s_3, s_4, p_2), \\ \hat{W} = (\hat{w}_1, \hat{w}_2, \dots, \hat{w}_8) = (w_1, w_2, \dots, w_5, s_5, s_6, p_3), \\ \hat{Z} = (\hat{z}_1, \hat{z}_2, \dots, \hat{z}_8) = (z_1, z_2, \dots, z_5, s_7, s_8, p_4). \end{cases}$$
(3)

The stego-block is shown in Fig. 3. Chang et al.'s scheme only uses five bits $(x_1, ..., x_5)$ in *X* as the input for f(x). Using five most significant bits in *X* as the input may degrade the visual quality of the stego-image for some cases.

2.5. Eslami et al.'s (k, n)-SAIS Scheme

Eslami et al. [7] employed a cellular automata (CA) method to construct a (k, n)-SAIS scheme. The polynomial-based (k, n)-SAIS scheme shares k secret pixels at each iteration with eight shared bits and then computes one authentication bit in [4,5] and four authentication bits in [6]. The CA-based (k, n)-SAIS scheme shares (k - 1) secret pixels at a time. Then the hash values of these (k - 1) secret pixels are jointly computed as eight bits for authentication. Afterwards, the k eightbit tuples are embedded by CA-based secret-sharing into eight bits



Fig. 3. The stego-block of Chang et al.'s (k, n)-SAIS scheme.



Fig. 4. The stego-block of Eslami et al.'s (k, n)-SAIS scheme.

 $(s_1,...,s_8)$ for each stego-block. The modified pixels \hat{X} , \hat{V} , \hat{W} , and \hat{Z} in the stego-block are shown in Eq. (4).

$$\begin{cases} X = (\hat{x}_1, \hat{x}_2, \dots, \hat{x}_8) = (x_1, x_2, \dots, x_6, s_1, s_2), \\ \hat{V} = (\hat{v}_1, \hat{v}_2, \dots, \hat{v}_8) = (v_1, v_2, \dots, v_6, s_3, s_4), \\ \hat{W} = (\hat{w}_1, \hat{w}_2, \dots, \hat{w}_8) = (w_1, w_2, \dots, w_6, s_5, s_6), \\ \hat{Z} = (\hat{z}_1, \hat{z}_2, \dots, \hat{z}_8) = (z_1, z_2, \dots, z_6, s_7, s_8). \end{cases}$$
(4)

The stego-block is shown in Fig. 4. These eight bits $(s_1,..., s_8)$ are not only share information bits but also include authentication information. The dealer signs the stego-image and other related information using his/her private key PR_D . Then, the signature and the corresponding public key PU_D are embedded into the stego-image for authentication. Finally, Eslami et al.'s scheme provides double authentication which one can use PU_D to verify whether a stego-image is tampered with in the first verification phase. In the second phase, one can check every *k* eight-bit tuples (k - 1 secret pixels and one authentication pixel) to detect tampered stego-blocks within the reconstructed information. Eslami et al.'s scheme requires fewer bits in each pixel to embed data, and thus enhance the visual quality of stego-image. Also, it employs cellular automata instead of Lagrange's polynomial to reduce the computational complexity from $O(n\log^2 n)$ to O(n).

3. Motivation and contribution

A SAIS scheme is often measured in terms of the visual guality of the stego-image through the peak-signal-to-noise ratio (PSNR) as well as in terms of authentication ability through the detection ratio of tampering (DR). However, good image quality and authentication ability have contradict characteristics. More authentication bits imply a higher DR; but this in turn reduces the PSNR. For example, Chang et al.'s scheme enhances authentication ability by using four authentication bits in a stego-block, but it degrades the stego-image quality. Alternatively, Eslami et al.'s scheme uses double authentication to reduce the number of authentication bits used in stegoimages. However, an invalid signature in Eslami et al.'s scheme only shows whether the stego-image is modified. If an attacker changes all bits in the stego-blocks but maintains the same eight shared bits (s_1, \dots, s_8) , we cannot locate the tampered stego-blocks. Although Eslami et al.'s scheme does not require authentication bit, its authentication ability is not exactly the same as Yang et al.'s scheme and Chang et al.'s scheme.

For achieving a high PSNR, it is reasonable to design a SAIS scheme that can locate tampered stego-blocks as the schemes in [5,6], but meantime does not require authentication bits. In this paper, we propose a vote-based SAIS scheme to provide authentication without authentication bits. All involved participants can mutually authenticate other participants. The validity of a stego-image is authenticated by a vote-based protocol. Every participant can authenticate other stego-images by voting for the stego-image that he/she trusts. Afterward, a majority vote on the authenticity of the stego-images is determined by the participants.

4. The Proposed (k, n)-SAIS Scheme

A (k-1)-degree univariate polynomial is used to construct (k, n)-SAIS schemes in [4–6]. Our (k, n)-SAIS scheme is based on a symmetric bivariate polynomial. We use the symmetric property to verify stego-images without using authentication bits.

4.1. Secret Sharing Scheme using Bivariate Polynomial

A bivariate polynomial is a polynomial with two variables. A (k-1)degree bivariate polynomial has the form $f(x, y) = \sum_{0 \le i,j \le (k-1)} a_{ij} x^i y^j$, on which some (k, n) verifiable secret sharing schemes and proactive secret sharing schemes were proposed [8–13]. Most of them used nonsymmetric bivariate polynomials to achieve different features of verifiable secret sharing schemes. In [9], the authors also showed an unconditionally secure (k, n) verifiable secret sharing scheme based on symmetric bivariate polynomial. A so-called bivariate polynomial f(x, y) of degree (k-1) is said to be symmetric if f(x, y) = f(y, x). This symmetric polynomial can be easily implemented by setting all coefficients a_{ij} and a_{ij} , $0 \le i$, $j \le (k-1)$, to be equal.

It can be easily verified that $f(x,y) = \sum_{0 \le i,j \le (k-1)} a_{ij}x^iy^j = \sum_{0 \le i,j \le (k-1)} a_{ji}x^iy^j = f(y,x)$ since $a_{ij} = a_{ji}$. In this paper, we use this symmetric bivariate polynomial in Eq. (5) to construct our (k, n)-SAIS scheme.

$$f(x,y) = \begin{pmatrix} a_{00} & a_{01}y & \dots & a_{0(k-1)}y^{k-1} \\ a_{01}x & a_{11}xy & \cdots & a_{1(k-1)}xy^{k-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{0(k-1)}x^{k-1} & a_{1(k-1)}x^{k-1}y & \cdots & a_{(k-1)(k-1)}x^{k-1}y^{k-1} \end{pmatrix} GF(2^8).$$
(5)

In Eq. (5), there are only 1+2+3+...+k=k(k+1)/2 distinct coefficients in f(x, y). Therefore, we can embed secret data into these distinct coefficients by sharing f(j, y), $1 \le j \le n$, to n stego-images where j is the identifier of participant j. From (5), the polynomial f(j, y) is reduced to $b_{j0}+b_{j1}y+...+b_{j(k-1)}y^{k-1}$.

This symmetric bivariate (k-1)-degree polynomial also has the same threshold property as Shamir's polynomial [9]. The threshold property implies that one can reconstruct f(x, y) from any k polynomials (say f(1, y), f(2, y), ..., f(k, y)). We then have the following k polynomials.

$$\begin{cases} f(1, y) = b_{10} + b_{11}y + \dots + b_{1(k-1)}y^{k-1}, \\ f(2, y) = b_{20} + b_{21}y + \dots + b_{2(k-1)}y^{k-1}, \\ \vdots \\ f(k, y) = b_{k0} + b_{k1}y + \dots + b_{k(k-1)}y^{k-1}. \end{cases}$$
(6)

By using the symmetric bivariate (k-1)-degree polynomial, we can encrypt k(k+1)/2 pixels into k shared pixels $(b_{j0}, b_{j1}, ..., b_{j(k-1)})$ for the *j*-th stego-image, $1 \le j \le n$. So, the shadow size is k/(k(k+1)/2) = 2/(k+1) times the secret image, which is different from 1/k by using Shamir's scheme. However, the symmetric property of bivariate polynomial can be used in our vote-based SAIS scheme to save the authentication bits.

4.2. The Encryption/Decryption Algorithm

The proposed (k, n)-SAIS scheme is based on symmetric bivariate polynomial f(x, y) of degree (k - 1). Every k(k + 1)/2 secret pixels are embedded into the coefficients in f(x, y). The k shared pixels are arranged carefully in k(k + 1)/2 stego-blocks for the j-th stego-image in order to obtain a better PSNR. On the other hand, the symmetric property is adopted to authenticate the validity of a stego-image mutually among all participants. Algorithm 1 and Algorithm 2 are encryption and decryption procedures, respectively. Description and diagrammatical representation of notations in our algorithms are defined below. Algorithm 1. Encryption of the proposed (*k*, *n*)-SAIS scheme

Input: *I*; O(j), $j \in [1, n]$; /* a secret image and *n* cover images */ **Output:** $\hat{O}(j)$, $j \in [1, n]$; /* *n* stego-images */

For i = 1 to $(W \times H)/(k(k+1)/2)$ do /* process every k(k+1)/2-pixeled unit at each iteration */

{ Gain *I*(*i*); /* obtain an unit from *I* */

{ For j = 1 to n do

{ Gain B(i, j); /* obtain an extended block from O(j) */ $F(I(i), B'(i, j)) = (s_1(i, j), ..., s_k(i, j))$; /* find k shared pixels (i.e., 8 k shared bits) */

Put back 8 *k* shared bits into the null LSBs in B'(i, j) to obtain a stegoextended block $\hat{B}(i, j)$;

/* As an example, for the proposed (3, *n*)-SAIS scheme, a stegoextended block (embracing 6 stego-blocks) is shown in Fig. 5. The binary form of a shared pixel is $s_1(i, j) = (s_{1, 1}(i, j), ..., s_{1, 8}(i, j))$. There are total 24 shared bits (i.e., 3 shared pixels) in (3, *n*)-SAIS scheme. For simplicity, Fig. 5 shows the structure of $\hat{B}(1, j)$. */

Put the stego- extended block $\hat{B}(i,j)$ on $\hat{O}(j)$;

}; /* end for *j* */

}; /* end for i */

The flow chart of our decryption is shown in Fig. 6. Any k stegoimages are first mutually verified by using the symmetric bivariate polynomial. Suppose that the k stego-images are all valid. Every stego-image should obtain (k-1) votes from others. Therefore, if any one stego-image gains less than (k-1) votes we stop reconstruction and proceed with the authentication procedure with the help of other (n-k) stego-image gains less than T votes from (n-1) participants we consider this stego-image fake; otherwise it is a valid stegoimage.

At this time, a majority vote is used to carry out the authentication. Majority rule is a decision rule that selects the valid stegoimages which have a majority (more than half the votes). It is reasonable to suppose more than half participants are honest. We identify whether the involved stego-images obtain a plurality of votes more numerous than manipulated stego-images. The number of the most votes from other participants is (n-1) votes. Therefore, a majority-voting threshold is chosen as T = [(n-1)/2]. Since $[(n-1)/2] = \ln/2$, it implies that we need a majority $\ln/2$ + 1 honest participants among all *n* participants to achieve the threshold. Obviously, the collusion of participants above a certain threshold can allow a fake stego-image to pass authentication. When choosing the threshold T = [(n-1)/2], our (k, n)-SAIS can tolerate $\ln/2$ colluded participants to compromise the authentication procedure.

Algorithm 2. Decryption of the proposed (*k*, *n*)-SAIS scheme

Input: Any *k* stego-images (say $\hat{O}(1)$, $\hat{O}(2)$, ..., $\hat{O}(k)$;

Output: *I*; /* verification of *k* stego-images and reconstruction of the secret image */

Phase 1–1 (Verification Phase: *k* participants involved in reconstruction):

For
$$i = 1$$
 to $(W \times H)/(k(k+1)/2)$ do

{For j = 1 to k do

{Every participant obtain polynomial f(A(i, j), y) from his/her own $\hat{B}(i, j)$;

k participants mutually check the validity of the $\hat{B}(i,j_1)$ and $\hat{B}(i,j_2)$ by $f(A(i,j_1), A(i,j_2)) = f(A(i,j_2), A(i,j_1))$, where $j_1, j_2 \in [1, k]$ and $j_1 \neq j_2$;

/* using the symmetric property of f(x, y) */ }; /* end for i */

$\hat{X}_{1}_{(x_{1,1},x_{1,2},,x_{1,7},\overline{s_{1,1}(1,j)})}$	$\hat{V}_{1} \\ (v_{1,1}, v_{1,2}, \dots, v_{1,7}, \overline{s_{1,2}(1,j)})$	$\hat{X}_{2}(x_{1,1}, x_{1,2},, x_{1,7}, \overline{s_{1,5}(1, j)})$	$ \hat{V}_{2} \\ (v_{1,1}, v_{1,2}, \dots, v_{1,7}, \overline{s_{1,6}(1, j)}) $
$(w_{1,1}, w_{1,2}, \dots, w_{1,7}, \boxed{s_{1,3}(1,j)})$	$\hat{Z}_{1} \\ (z_{1,1}, z_{1,2},, z_{1,7}, \overline{s_{1,4}(1, j)})$	$ \hat{W}_{2} \\ (w_{1,1}, w_{1,2}, \dots, w_{1,7}, \overline{s_{1,7}(1,j)}) $	$\hat{Z}_{2}(z_{1,1}, z_{1,2},, z_{1,7}, \overline{s_{1,8}(1, j)})$
$\hat{X}_{3}_{(x_{1,1},x_{1,2},,x_{1,7},s_{2,1}(1,j))})$	$\hat{V}_{3} \\ (v_{1,1}, v_{1,2}, \dots, v_{1,7}, \overline{s_{2,2}(1, j)})$	$\hat{X}_{4}_{(x_{1,1},x_{1,2},,x_{1,7},\overline{s_{2,5}(1,j)})}$	$\hat{V}_{4} \\ (v_{1,1}, v_{1,2}, \dots, v_{1,7}, \overline{s_{2,6}(1,j)})$
$(w_{1,1}, w_{1,2}, \dots, w_{1,7}, \overline{s_{2,3}(1,j)})$	$\hat{Z}_{3}(z_{1,1}, z_{1,2},, z_{1,7}, s_{2,4}(1, j)))$	$(w_{1,1}, w_{1,2}, \dots, w_{1,7}, \overline{s_{2,7}(1,j)})$	$ \begin{array}{c} \hat{Z}_{4} \\ (z_{1,1}, z_{1,2}, \dots, z_{1,7}, \boxed{s_{2,8}(1, j)}) \end{array} $
$\hat{X}_{5}(x_{1,1}, x_{1,2},, x_{1,7}, \overline{s_{3,1}(1,j)})$	$\hat{V}_{5}(v_{1,1}, v_{1,2},, v_{1,7}, \overline{s_{3,2}(1, j)})$	$\hat{X}_{6} \\ (x_{1,1}, x_{1,2},, x_{1,7}, \overline{s_{3,5}(1,j)})$	$\hat{V}_{6} \\ (v_{1,1}, v_{1,2}, \dots, v_{1,7}, \overline{s_{3,6}(1, j)})$
$(w_{1,1}, w_{1,2},, w_{1,7}, \overline{s_{3,3}(1,j)})$	$\hat{Z}_{5}(z_{1,1}, z_{1,2},, z_{1,7}, \overline{s_{3,4}(1, j)})$	$ \begin{bmatrix} \hat{W}_{6} \\ (w_{1,1}, w_{1,2},, w_{1,7}, \boxed{s_{3,7}(1, j)} \end{bmatrix} $	$\hat{Z}_{6} \\ (z_{1,1}, z_{1,2},, z_{1,7}, \overline{s_{3,8}(1, j)})$

Fig. 5. Arrange 24 shared bits (3 shared pixels: $s_1(1, j), s_2(1, j), s_3(1, j)$) into $\hat{B}(1, j)$ uniformly for the proposed (3, *n*)-SAIS scheme.

}; /* end for *j* */

If any stego-image gain less than (k-1) votes then go to **Phase 1–2**;

Else {k stego-images are valid; go to **Phase 2**};

Phase 1–2 (Verification Phase: all *n* **involved participants):** /* (1) stop reconstruction

(2) add other (n-k) stego-images to verify these k stego-images */

Other (n - k) participants obtain polynomial f(A(i, j), y) from his/her own $\hat{B}(i, j)$;

(n-k) participants verify these k stego-images by $f(A(i, j_1), A(i, j_2)) = f(A(i, j_2), A(i, j_1))$, where $j_1 \in [k+1, n]$ and $j_2 \in [1, k]$;

If any stego-image gain less than *T* votes then the stego-image is fake;

Else the stego-image is valid;

/* (1) a majority threshold $T = \lceil (n-1)/2 \rceil$ is used in the vote-based scheme



Fig. 6. The flow chart of decryption for the proposed (k, n)-SAIS scheme.

(2) our (*k*, *n*)-SAIS can tolerate $\lfloor n/2 \rfloor$ colluded participants to compromise the authentication */

Stop reconstruction;

Phase 2 (Reconstruction Phase):

For i = 1 to $(W \times H)/(k(k+1)/2)$ do

{For j = 1 to k reconstruct f(A(i, j), y) from his/her own $\hat{B}(i,j)$ };

Derive a bivariate (k-1)-degree f(x, y) from k polynomials f(A(i, 1), y), ..., f(A(i, k), y);

Obtain the unit I(i) from all k(k+1)/2 coefficients in f(x, y); }; /* end for $i^*/$

Reconstruct a secret image *I* from all units *I*(1), *I*(2), ..., and $I\left(\frac{W \times H}{k(k+1)/2}\right)$;

Example 1. For the proposed (3, 3)-SAIS scheme, we embed 6 (=k(k+1)/2) secret pixels into one stego-extended block (6 stego-blocks) for each stego-image.

The generation polynomial used in $GF(2^8)$ is $x^8 + x^4 + x^3 + x^2 + 1$. By embedding these six secret pixels into six distinct coefficients in f(x, y), we have the following symmetric bivariate polynomial f(x, y), as shown in Eq. (7) (also see Fig. 7(a-2)).

$$f(x,y) = 28 + 123y + 56y^{2} + 123x + 49xy + 88xy^{2} + 56x^{2} + 88x^{2}y + 95x^{2}y^{2}.$$
(7)

Given $A(i, j) = H_K((B'(i, j)) \|i\| \|j\| W\| H)_8$, suppose that A(i, 1) = 48, A(i, 2) = 184, and A(i, 3) = 90. For stego- images #1, #2 and #3, the following three polynomials f(A(i, j), y), j = 1, 2, 3, are shown in Eq. (8) (see Fig. 7(a-3)).

$$\begin{cases} f(48,y) = 52 + 204y + 242y^2, \\ f(184,y) = 201 + 126y + 53y^2, \\ f(90,y) = 75 + 232y + 92y^2. \end{cases}$$
(8)

According to Fig. 5, we arrange 24 shared bits (three shared pixels $s_1(i, 1)=52$, $s_2(i, 1)=204$, and $s_3(i, 1)=242$ for stego-image #1; three shared pixels $s_1(i, 2)=201$, $s_2(i, 2)=126$, and $s_3(i, 2)=53$ for stego-image #2; three shared pixels $s_1(i, 3)=75$, $s_2(i, 3)=232$,

C.-N. Yang et al. / Optics Communications 285 (2012) 1725–1735



Fig. 7. Generation of stego-blocks for the proposed (3, 3)-SAIS scheme: (a) six secret pixels, f(x, y), and f(A(i, j), y), j = 1, 2, 3 (b) stego-image #1 (c) stego-image #2 (d) stego-image #3.

and $s_3(i, 3) = 92$ for stego-image #3) into one stego-extended block (24 pixels). Fig. 7(b-d) shows 24 pixels in cover image and the modified pixels in stego-image, and show how to embed the 24 shared bits.

Here, we demonstrate that every participant can authenticate the validity of other stego-images. For example, participant #1 can derive f(48, y), f(184, y), f(90, y), and A(i, 1) = 48, A(i, 2) = 184, and A(i, 3) = 90 from his/her stego-image and other two stego-images. By verifying f(48, 184) = f(184, 48) = 174 and f(48, 90) = f(90, 48) = 118, he/she can successfully authenticate stego-images #2 and #3. Consider the case in which a stego-extended block is tampered with (say $\hat{B}(i, 2)$). There are three possible cases for tampering with $\hat{B}(i, 2)$ as follows.

Case (1) modified bits are in B'(i, 2) and do not include the shared bits.

In this case, the polynomial f(184, y) is not modified since the coefficients of f(184, y) are obtained from the shared bits of $s_1(i, 2)$, $s_2(i, 2)$, and $s_3(i, 2)$. However, the modifications in B'(i, 2) result in A'(i, 2), which equals A(i, 2) with the probability 1/256. So, f(48, A'(i, 2)) only has the same value of f(148,48) = 174 with a probability of 1/256 to pass authentication. For example, if A'(i, 2)=183, then f(48, 183) = 14, and thus $f(48, 183) \neq f(184, 48)$. Participant #1 then successfully detects tampering in stego-image #2.

Case (2) modified bits are located in the shared bits of $s_1(i, 2)$, $s_2(i, 2)$, and $s_3(i, 2)$.

In this case, the polynomial f(184, y) is modified to f'(184, y), while the value of A(i, 2) = 184 is not changed. Suppose only one shared bit $s_{1, 1}(i, 2)$. Thus, we have $s_{1, 1}(i, 2)=73$, and then $f'(184, y)=73+126y+53y^2$. Since f'(184, 48)=

46, participant #1 finds that $f'(184, 48) \neq f(48, 84)$ and then detects tampering.

Case (3) modified bits are located in both B'(i, 2) and the shared bits $(s_1(i, 2), s_2(i, 2), s_3(i, 2))$. In this case, the polynomial f(184, y) and the value of A(i, 2) are both changed to f'(184, y) and A'(i, 2), respectively. f(48, A'(i, 2)) and f'(184, A(i, 1)) have the same value with probability 1/256. For example, if $f'(184, y)=73 + 126y+53y^2$ and A'(i, 2)=183, then f(48, 183)=46 and f'(184, 48)=46. Since they are not equal, tampering is

If every shadow gain two votes from other two participants, the stego-images are authenticated successfully. We then proceed to reconstruct f(x, y) from f(48, y), f(184, y) and f(90, y). Suppose we only have two polynomials (say f(48, y) and f(184, y)). From f(48, y) and f(184, y), we could derive six equations in Eq. (9). One can easily verify the rank of these six equations is only 5. Actually, we cannot solve these equations for six variables. By adding Eq. (10) from f(90, y), we can solve $a_{00} = 28$, $a_{01} = 123$, $a_{02} = 56$, $a_{11} = 49$, $a_{12} = 88$, and $a_{22} = 95$ from Eqs. (9) and (10). Finally, we recover six secret pixels.

detected.

 $\begin{pmatrix} a_{00} + 48 \times a_{01} + 48^2 \times a_{02} = 52 \Rightarrow a_{00} + 48 \times a_{01} + 105 \times a_{02} = 52 \\ a_{01} + 48 \times a_{11} + 48^2 \times a_{12} = 204 \Rightarrow a_{01} + 48 \times a_{11} + 105 \times a_{12} = 204 \\ a_{02} + 48 \times a_{12} + 48^2 \times a_{22} = 242 \Rightarrow a_{02} + 48 \times a_{12} + 105 \times a_{22} = 242 \\ a_{00} + 184 \times a_{01} + 184^2 \times a_{02} = 201 \Rightarrow a_{00} + 184 \times a_{01} + 58 \times a_{02} = 201 \\ a_{01} + 184 \times a_{11} + 184^2 \times a_{12} = 126 \Rightarrow a_{01} + 184 \times a_{11} + 58 \times a_{12} = 126 \\ a_{02} + 184 \times a_{12} + 184^2 \times a_{22} = 53 \Rightarrow a_{02} + 184 \times a_{12} + 58 \times a_{22} = 53$



Fig. 8. Modified LSBs in (3, 5)-SAIS scheme: (a) Lin et al.'s scheme: 3 stego-blocks have 1 MB and 2 AB (b) Yang et al.'s scheme: 3 stego-blocks have 1 MB and 2 AB (c) Chang et al.'s scheme: 3 stego-blocks have 1 MB and 2 AB (d) Eslami et al.'s scheme: 2 stego-blocks have 1 SB and 1 NB (e) the proposed scheme: 6 stego-blocks have 6 SB.

$$\begin{cases} a_{00} + 90 \times a_{01} + 90^2 \times a_{02} = 75 \Rightarrow a_{00} + 90 \times a_{01} + 148 \times a_{02} = 75 \\ a_{01} + 90 \times a_{11} + 90^2 \times a_{12} = 232 \Rightarrow a_{01} + 90 \times a_{11} + 148 \times a_{12} = 232 \\ a_{02} + 90 \times a_{12} + 90^2 \times a_{22} = 92 \Rightarrow a_{02} + 90 \times a_{12} + 148 \times a_{22} = 92 \end{cases}$$
(10)

5. Evaluation and experiment

5.1. Analysis of the PSNR and DR

It is observed that the schemes in [4-6] share every k secret pixels into one shared pixel; Eslami et al.'s SAIS scheme shares every (k-1) secret pixels into one shared pixel, and our scheme embeds k(k+1)/2 secret pixels into k shared pixels. All SAIS schemes [4-7] and our scheme have a stego-image size four times

that of the secret image. We now briefly describe the stego-blocks in the schemes in [4–7] and the stego- extended block in our scheme.

The shared block (SB), the authenticated block (AB), the mixed block (MB), and the non-embedded block (NB) are defined as follows. SB is a block that only has shared bits, and AB is a block that only has authentication bits. MB has both shared bits and authentication bits, while NB does not embed any information. The aim of NB is only to keep the stego-image size four times that of the secret image. Since the schemes in [4–6] have one MB and (k-1) AB ((refer to Fig. 4 in [14])). Also, Lin et al.'s scheme and Yang et al.'s scheme both have 9 modified bits in one MB and 1 modified bit in one AB. However, Chang et al.'s scheme has 12 modified bits in

(a) Jet

(b-1) Lena: 52.91 dB

(b-2) Pepper: 52.91 dB

(b-3) Baboon: 52.91 dB



(b-4) Elaine: 52.90 dB



(b-5) Boat: 52.91 dB



Fig. 9. The reconstructed images and five stego-images for the proposed (3, 5)-SAIS scheme:(a) a 256×256-pixeled secret image (b) five 512×512-pixeled stego-images.

Table 1

notation	description	diagrammatical representation
I I(i)	<i>I</i> is the secret image with $(W \times H)$ pixels, which is divided into $I(i)$ units, $1 \le i \le (W \times H)/(k(k+1)/2)$. Every unit $I(i)$ has $k(k+1)/2$ pixels.	$I \underbrace{I(1) I(2)}_{I(i)} \\ \downarrow \\ k(k+1)/2 \text{ pixels}$
O(j) Ô(j)	$O(j)$, $1 \le j \le n$, is the <i>j</i> -th cover image with $(2 W \times 2H)$ pixels. There are $(W \times H)$ blocks. Every block has four pixels X_i , W_i , V_i and U_i , $1 \le i \le (W \times H)$. Suppose that x_i is the pixel value for X_i and its binary representation is $(x_{i, 1}, x_{i, 2},, x_{i, 8})$. Other notations w_i , v_i and u_i follow similarly. On the other hand, $\hat{O}(j)$ is the <i>j</i> -th stego-image after embedding the secret image. Every stego-block has four pixels \hat{X}_i , \hat{W}_i , \hat{V}_i and \hat{U}_i .	$ \begin{array}{c} O(j) \\ X_1 \\ W_1 \\ U_1 \\ W_2 \\ U_3 \\ \hline W_1 \\ U_1 \\ \hline W_1 \\ U_3 \\ \hline U_3 \\ \hline U_3 \\ \hline U_4 \\ \hline U_4$
$B(i,j) \ \hat{B}(i,j)$	$B(i, j), 1 \le i \le (W \times H)/(k(k+1)/2)$, is the <i>i</i> -th extended block in $O(j)$. Every extended block has $k(k+1)/2$ blocks. On the other hand, $\hat{B}(i, j)$ is the <i>i</i> -th stego-extended block in $\hat{O}(j)$. Every stego-extended block has $k(k+1)/2$ stego-blocks.	$\begin{array}{c c c c c c c c c c c c c c c c c c c $
B'(i, j)	<i>B'</i> (<i>i</i> , <i>j</i>) are the remaining bits which 8 <i>k</i> least significant bits are excluded from <i>B</i> (<i>i</i> , <i>j</i>).	$B(i, j)$ $X_{l+1} V_{l+1}$ $W_{l+1} U_{l+1}$ remove 8k least significant bits (LSBs) uniformly from $k(k+1)/2$ blocks. $B'(i, j)$ $X_{l+1} V_{l+1}$ $W_{l+1} U_{l+1}$
F(·,·)	By using the unit $l(i)$ $(k(k+1)/2$ pixels) as the $k(k+1)/2$ distinct coefficients in a bivariate polynomial to generate a $(k-1)$ -degree symmetric polynomial $f(x, y)$. Then, we gain a pixel $A(i, j) = H_k((B'(i, j)) \ i\ j\ W\ H)_{B_k}$, which is an eight-bit hash result of $B'(i, j)$, the ex- tended block ID i'i', the stego-image ID "j", and the secret image size W and H . Suppose that $H_k(\cdot)$ is a keyed hash function, e.g., HMAC defined in RFC2104 requiring a secret K for operation. Afterwards, we use $A(i, j)$ in substitution for x in $f(x, y)$ to calculate $f(A(i, j), y) =$ $s_1(i, j) + s_2(i, j)y + + s_k(i, j)y^{k-1} = b_{k-1}$. Finally, we get 8 k shared bits $F(I(i), B'(i, j)) =$ $(s_1(i, j),, s_k(i, j))$.	$B'(i, j)$ $\overline{X_{l+1}} V_{l+1} \dots \dots W_{l+1} U_{l+1} \dots \dots \dots W_{l+1} U_{l+1} \dots \dots \dots \dots W_{l+1} U_{l+1} \dots $

one MB and 4 modified bits in one AB. Eslami et al.'s scheme has 8 modified bits in one SB and (k-2) NB. The proposed scheme embeds 8 k shared bits into k(k + 1)/2 stego-blocks. So, it has k(k + 1)/22 SB.

From the above description, we can easily derive the average modified bits per stego-block for all (k, n)-SAIS schemes. Let the number of average modified bits per stego-block for Lin et al.'s scheme, Yang et al.'s scheme, Chang et al.'s scheme, Eslami et al.'s scheme, and the proposed scheme be N_{LIN} , N_{YAN} , N_{CHA} , N_{ESL} , and N_{PRO} , respectively. All these values are derived as follows.

$$\begin{cases} N_{LIN} = N_{YAN} = (1 \times 9 + (k-1) \times 1)/k = 1 + (8/k), \\ N_{CHA} = (1 \times 12 + (k-1) \times 4)/k = 4 + (8/k), \\ N_{ESL} = (1 \times 8 + (k-2) \times 0)/(k-1) = 8/(k-1), \\ N_{PRO} = (k \times 8)/(k \times (k+1)/2) = 16/(k+1). \end{cases}$$
(11)

Since 6/(k+1) < 4 + (8/k), $N_{PRO} < N_{LIN}$ and $N_{PRO} < N_{YAN}$. From the values of 16/(k+1) and 1 + (8/k), we have $N_{PRO} \approx N_{CHA}$ for $2 \le k \le 5$, and $N_{PRO} < N_{CHA}$ when k > 6. Also, we have $N_{PRO} \le N_{ESL}$ for k = 2, 3. Eslami et al.'s scheme has the smallest value 8/(k-1) among these schemes. The fewer modified LSBs cause the higher PSNR of stego-image. Our modified bits are arranged uniformly across all stego-blocks to achieve the better image quality. Experimental results show that the proposed (3, 5)-SAIS scheme has the best PSNR among all (3, 5)-SAIS schemes considered here.

Next, we evaluate authentication ability and estimate DR for all SAIS schemes. The authentication bit in Lin at.'s scheme is simply chosen to make this pixel even or odd parity as a binary parity sequence generated by a secret key. Because a participant can determine parity information from his/her own stego-image, he/she can maliciously make a fake stego-image to pass authentication. So, Lin et al.'s scheme does not have the authentication ability to prevent dishonest participants from malicious modifications as shown in [5]. Eslami et al.'s scheme provides double authentication. One can verify the signature of a stego-image to assure its integrity by using PU_D. However, if an attacker changes the bits in every (k-1) stego-blocks but maintains the eight shared bits $(s_1, ..., s_8)$ as constant, tampering cannot be detected. In this case, an invalid signature only shows that the stego-image is modified, but we cannot locate the tampered stegoblocks. The authentication ability of Eslami et al.'s scheme is not exactly the same as Yang et al.'s and Chang et al.' schemes, both of which can locate tampered stego-blocks in stego-images, as shown in Figs. 6, 7 and 8 in [5] and Fig. 9 in [6]. However, Eslami et al.'s scheme only detects the invalidity of shared bits with probability 255/256, while it cannot detect whether tampering occurred beyond shared bits. So, the DRs for the schemes by Lin et al., Yang et al., Chang et al. and Eslami et al. are estimated as 0, 1/2 (i.e., one authentication bit), 15/16 (i.e., four authentication bits), and 0, respectively.

Our scheme simultaneously uses shared bits for reconstruction and authentication. We derive f(A(i, j), y) from stego-images, and then use the symmetric property to mutually authenticate stegoblocks by $f(A(i, j_1), A(i, j_2)) = f(A(i, j_2), A(i, j_1))$, where $j_1 \neq j_2$. The inputs of $A(i, j_1)$ and $A(i, j_2)$ are obtained from all bits in $B'(i, j_1)$ and $B'(i, j_2)$. So, an attacker or dishonest participant cannot exactly derive the input without the secret key K. Since the outputs of $f(A(i, j_1), A(i, j_2))$ and $f(A(i, j_2), A(i, j_1))$ are both eight bits, authentication will fail with probability 255/256. Yang et al.'s scheme and Chang et al.'s scheme have DR = 1/2 and DR = 15/16 for a stego-block, respectively. The propose scheme has DR = 255/266 for a stego-extended block. Although our DR is higher than Yang et al.'s and Chang et al.'s DR, the detected block size is a stego-extended block of k(k+1)/2stego-blocks. Yang et al.'s scheme and Chang et al.'s scheme have a more precise detection area of one stego-block.

5.2. Experimental result and comparison

Using a (3, 5)-SAIS scheme as an example, the schemes in [4–6] process three blocks at each iteration, and Eslami et al.'s scheme [7] processes every two blocks. Meanwhile, our scheme processes an extended block (6 blocks) at each iteration. The modified LSBs in stegoblocks for all schemes are shown in Fig. 8, where the number in the rectangle is the number of modified bits in a pixel. Lin et al.'s scheme, Yang et al.'s scheme and Chang et al.'s scheme have one MB and two

AB. Eslami et al.'s scheme has one SB and one NB. Our scheme has 6 SB.

From Fig. 8, Lin et al.'s (3, 5)-SAIS scheme shows an average number of modified bits per stego-block of $N_{UN} = (9+1+1)/3 = 3.6$ bits. Although Yang et al.'s (3, 5)-SAIS scheme has the same number of modified bits (i.e., $N_{YAN} = 3.6$ bits), the arrangement (2-3-2-2) is more "flat" than (0-3-3-3) and thus yields a better PSNR. Chang et al.'s (3, 5)-SAIS scheme and Eslami et al.'s (3, 5)-SAIS scheme result in $N_{CHA} = 6.6$ bits and $N_{ESL} = 4$ bits, respectively. The proposed (3, 5)-SAIS scheme also has $N_{PRO} = 4$ bits, and all modified LSBs are arranged uniformly across all 6 stego-blocks. All PSNRs of the stegoimages for these five (3, 5)-SAIS scheme are shown in Table 1. Five images (namely, Lena, Pepper, Baboon, Elaine, and Boat) are used as cover images, and let is used as the secret image. Notice that we should embed an additional 2304 bits (i.e., a public key of 2048 bits and a digital signature of 256 bits) into the stego-images for Eslami et al.'s scheme. It is observed that the proposed (3, 5)-SAIS scheme has the best result among all schemes. When compared with Chang et al.'s (3, 5)-SAIS scheme, our (3, 5)-SAIS scheme even enhances the quality of the stego-images by about 10 dB.

The PSNR of a stego-image for the proposed (k, n)-SAIS scheme, where $k \ge 3$, can be estimated as follows. We first calculate the average mean square error of a stego-image under our (k, n)-SAIS scheme. Let N be the average modified bits per pixel. Since $N_{PRO} = 16/(k+1)$, so N = 4/(k+1). Recalling that a stego-block has four pixels, thus we have $N \le 1$ for $k \ge 3$. This observation implies that the visual quality of the stego-image will be degraded only by LSB for our (k, n)-SAIS scheme, where $k \ge 3$. All pixels in a stego-image have the probability 4/(k+1) of changing their respective LSB. Let the pixel difference be $v, v \in [-1, 0, +1]$, which is uniformly distributed with probability f(v)dv. Since f(v) is uniformly distributed and thus can be considered a constant, f = 1/3, in the difference interval. The average mean square error of a stego-image is $4/(k+1) \times \left(\sum_{\nu \in [-1,0,+1]} f(\nu) \times \nu^2\right) = \left(4/(3 \times (k+1)) \times \sum_{\nu \in [-1,0,+1]} \nu^2 = 8/(3 \times (k+1)).$ Hence, the estimated PSNR is PSNR_{est}= $10 \times \log_{10}(255^2/(8/(3 \times (k+1))))$ dB. For k=3, $PSNR_{est} = 49.89 \text{ dB}$, which approximates the values in Table 2. Five stego-images for our (3, 5)-SAIS scheme and the reconstructed secret image are shown in Fig. 9.

To evaluate authentication ability, we focus on Yang et al.'s scheme, Chang et al.'s scheme, and the proposed scheme. The other two schemes with DR = 0 either cannot avoid dishonest participants (i.e., Lin et al.'s scheme) or cannot locate the tampered blocks in some cases (i.e., Eslami et al.'s scheme). For the proposed (3, 5)-SAIS scheme, we intentionally counterfeit a fake stego-image. As shown in Fig. 10(a), a shrunk version of Lena is added to the upper left corner in the Lena stego-image. Fig. 10(b) is the authentication result for the proposed scheme. By the same manipulation, the authentication results for Yang et al.'s scheme and Chang et al.'s scheme are shown in Figs. 10(c) and (d). All localization areas in Fig. 9(b-d) shows the rectangular shape of places that were tampered with. The black color in the tampered area denotes the detection of manipulation, while the other areas pass authentication. The proposed scheme shows DR \approx 255/256, and thus, the tampered area is almost entirely black. However, our detected unit is a stegoextended block (6 stego-blocks or 24 pixels) and thus cannot precisely indicate the tampered places. Fig. 10 (c) and (d) has the same detected unit of a stego-block (4 pixels), and $DR \approx 1/2$ for Yang et al.'s scheme is lower than $DR \approx 15/16$ for Chang et al.'s scheme. So, there are fewer black stego-blocks in Fig. 10(c), which one can still reveal the shrunken Lena. This implies that an attacker has a greater possibility to generate a fake stego-image and pass authentication under Yang et al.'s scheme.

Based on the above experimental results, Table 3 summarizes the comparison of the (k, n)-SAIS schemes for the following items: (1) the expansion of the stego-image, (2) the authentication capability,

Table 2				
PSNRs of stego-images	for f	ive (3,	5)-SAIS	schemes.

_						
		Lin et al.'s scheme	Yang et al.'s scheme	Chang et al.'s Scheme	Eslami et al.'s scheme	The proposed scheme
	Lena Pepper Baboon Elaine	43.82 dB 43.78 dB 43.81 dB 43.77 dB	46.11 dB 46.14 dB 46.12 dB	42.28 dB 42.30 dB 42.31 dB 42.29 dB	47.59 dB 47.53 dB 47.55 dB 47.50 dB	52.91 dB 52.91 dB 52.91 dB 52.90 dB
	BOat	43.80 dB	46.10 dB	42.22 dB	47.51 dB	52.91 dB
	Lena Pepper Baboon Elaine Boat	43.82 dB 43.78 dB 43.81 dB 43.77 dB 43.80 dB	46.11 dB 46.14 dB 46.14 dB 46.12 dB 46.10 dB	42.28 dB 42.30 dB 42.31 dB 42.29 dB 42.22 dB	47.59 dB 47.53 dB 47.55 dB 47.50 dB 47.51 dB	52.91 dB 52.91 dB 52.91 dB 52.90 dB 52.90 dB 52.91 dB

(3) the size of the detection unit, (4) the authentication bits in a stego-block, (5) the average number of modified bits per stego-block, and (6) the number of processed stego- blocks.

To consist with the first SAIS scheme [4], all schemes use a stegoimage four times larger than the secret image, even though some schemes do not need to expand their stego-images by four times. For example, when discarding the AB, the stego-image size in [4–6] can be reduced by 4/k times of the secret image because AB is not required in them in order to share secret pixels. Considering authentication capability, Lin et al.'s scheme is compromised by the dishonest participant problem. As shown in [4], a dishonest participant can perform three types of manipulations, namely, the unobvious modification of the stego-image, the obvious modification of the stego-image, and the replacement of the stego-image. Any modification of stego-image in Eslami et al.'s scheme is detected by verifying the signature, but it only detects the invalidity of SB with probability 255/256 and cannot locate modification areas if the tampering occurs at NB. Because of removing authentication bits from the stego- extended block, our scheme only coarsely indicates tampered areas with the size of k(k+1)/2 stego-blocks. In the proposed SAIS scheme, it is possible to reduce the size of detection unit by changing some SB to MB. However, our small PSNR is due to the fact that authentication bits are not required. The changing some SB to MB apparently will degrade the PSNR of stego-images. Our scheme enhances the PSNR at the cost of increasing the detection size. All in all, form Table 3, our SAIS scheme outperforms other SAIS schemes.

6. Conclusion

In this paper, we discuss the visual quality of stego-image (i.e., the PSNR) and authentication ability (i.e., DR) of a (k, n)-SAIS scheme. A (k, n)-SAIS scheme is proposed based on symmetric bivariate polynomial. We combine both authentication and secret sharing features into shared bits without needing additional authentication bits. The validity of a stego-image is authenticated by a vote-based protocol. Our (k, n)-SAIS scheme achieves high PSNR and DR as compared with existing SAIS schemes.

Acknowledgment

This work was supported in part by the National Science Council project under Grant NSC 99-2631-H-259-001-, and the Testbed@TWISC, National Science Council under the Grant NSC 100-2219-E-006-001.



Fig. 10. Authentication results: (a) a fake stego-image (b) authentication of the proposed scheme (c) authentication of Yang et al.'s scheme (d) authentication of Chang et al.'s scheme.

Table 3

Comparison of (k, n)-SAIS schemes.

		Lin et al.'s scheme	Yang et al.'s scheme	Chang et al.'s Scheme	Eslami et al.'s Scheme	The proposed scheme
expansion of stego-image		4	4	4	4	4
authentication capability		DR = 0	DR = 1/2	DR = 15/16	DR = 0	DR = 255/256
the size of detection unit		_	1 stego-block	1 stego-block	—	k(k+1)/2 stego-blocks
authentication bits in a stego-block		1	1	4	0	0
average number of modified bits	k	1 + (8/k)	1 + (8/k)	4 + (8/k)	8/(k-1)	16/(k+1)
per stego-block	2	5	5	8	8	5.33
	3	3.67	3.67	6.67	4	4
	4	3	3	6	2.67	3.2
	5	2.6	2.6	5.6	2	2.66
	6	2.33	2.33	5.33	1.6	2.28
the number of processed stego-	k	k stego-blocks (1 MB, (k	k stego-blocks (1 MB, (k	k stego-blocks (1 MB, (k	(k-1) stego-blocks (1 SB,	k(k+1)/2 stego-blocks
blocks each time		-1) AB)	-1) AB)	-1) AB)	(k-2) NB)	(k(k+1)/2 SB)
	2	1 MB, 1 AB	1 MB, 1 AB	1 MB, 1 AB	1 SB, 0 NB	3 SB
	3	1 MB, 2 AB	1 MB, 2 AB	1 MB, 2 AB	1 SB, 1 NB	6 SB
	4	1 MB, 3 AB	1 MB, 3 AB	1 MB, 3 AB	1 SB, 2 NB	10 SB
	5	1 MB, 4 AB	1 MB, 4 AB	1 MB, 4 AB	1 SB, 3 NB	15 SB
	6	1 MB, 5 AB	1 MB, 5 AB	1 MB, 5 AB	1 SB, 4 NB	21 SB

References

- [1] A. Shamir, Communications of the Association for Computing Machinery 22 (1979) 612.

- (1979) 612.
 [2] C.C. Thien, J.C. Lin, Computer and Graphics 26 (2002) 765.
 [3] R.Z. Wang, C.H. Su, Pattern Recognition Letters 27 (2006) 551.
 [4] C.C. Lin, W.H. Tsai, Journal of Systems and Software 73 (2004) 405.
 [5] C.N. Yang, T.S. Chen, K.H. Yu, C.C. Wang, Journal of Systems and Software 80 (2007) 1070.
 [6] C.C. Chang, Y.P. Hsieh, C.H. Lin, Pattern Recognition 41 (2008) 3130.
 [7] Z. Eslami, S.H. Razzaghi, J.Z. Ahmadabadi, Pattern Recognition 43 (2010) 397.
 [8] R. Gennaro, Y. Ishai, E. Kushilevitz, T. Rabin, Proceedings of STOC, 2001, p. 580.

- [9] V. Nikov, S. Nikova, Lecture Notes in Computer Science 3357 (2005) 308.
 [10] M. Fitzi, J. Garay, S. Gollakota, C. Pandu Rangan, K. Srinathan, Lecture Notes in Computer Science 3876 (2006) 329.
- [11] J. Katz, C. Koo, R. Kumaresan, Lecture Notes in Computer Science 5126 (2008) , 499.
- [12] A. Patra, A. Choudhary, T. Rabin, C. Pandu Rangan, Lecture Notes in Computer Science
- [12] M. Tata, A. Choudhai, Y. Kabin, C. Pandu Rangan, Eccure Notes in Computer Science 6477 (2009) 487.
 [13] R. Kumaresan, A. Patra, C. Pandu Rangan, Lecture Notes in Computer Science 6477 (2010) 431.
- [14] C.N. Yang, C.B. Ciou, Pattern Recognition 42 (2009) 1615.