Pattern Recognition Letters 34 (2013) 283–291

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

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

Secret image sharing scheme with adaptive authentication strength

Guzin Ulutas *, Mustafa Ulutas, Vasif V. Nabiyev

Department of Computer Engineering, Karadeniz Technical University, 61080 Trabzon, Turkey

ARTICLE INFO

Article history:
Received 9 July 2011
Available online 10 November 2012

Communicated by G. Borgefors

Keywords:
Secret sharing
Steganography
Chaining
Authentication

ABSTRACT

Transmission of secret messages or images over the Internet using Shamir's secret sharing scheme has become popular. Some researchers use steganography with Shamir's method to hide noise like share images into natural looking cover images to improve secrecy. Stego images are authenticated against accidental or deliberate changes before recovering the secret. Authentication by a parity bit stream calculated by a keyed hash of stego images is commonly used. Researchers aim to increase the number of authentication bits to improve the authentication strength of their methods. Eslami and Ahmadabadi (2011) proposed a method with dynamic embedding strategy in 2011. They use a concatenated string of four bits, two from the current and two from previous block, to authenticate individual blocks. Even though chaining performs block based authentication, it cannot detect individual fake stego blocks and cannot authenticate the rest of the stego image blocks if it faces a changed block. This paper proposes a new secret image sharing method by selecting the number of authentication bits proportional to block size, contrary to Eslami and Ahmadabadi (2011) method which uses four bits to authenticate blocks regardless of block size. The proposed method has improved authentication for increased block size and can authenticate individual stego blocks as well. It produces good quality stego images and can still authenticate the rest of the stego image even after an altered stego block is encountered as shown in the experimental results.

1. Introduction

The use of mature networking standards and technologies make data transmission faster, reliable and popular among a large number of wired and wireless applications. The broadcast nature of wired and wireless networks needs secure transmission techniques to ensure the secrecy of sensitive data such as military or commercial images. Two techniques used for secure transmission are cryptography and steganography. Cryptography maps secret data into unreadable form by mathematical transformations. The noise like nature of the cipher data draws attention of eavesdroppers. Therefore, latter technique is proposed to hide the secret data into another medium called cover. Cover medium can be a natural looking digital data such as an image, video, or even a text file. Cover medium that holds secret data after embedding procedure is called stego medium. However both methods are vulnerable to loss or damage of cipher or stego file since the secret cannot be revealed.

Researchers have proposed threshold based secret sharing schemes to overcome lost or damaged file problem. Threshold schemes of (k,n) share a secret among n participants where each participant gets a piece of information called a share. Even though each share holds some information to overcome the lost, it is not possible to reveal the secret unless any k or more of the shares gather. Shamir (1979) proposed a threshold scheme based on polynomial interpolation in 1979. His method forms a $(k-1)$th degree polynomial. The polynomial is evaluated for unique x values and these are distributed to the participants. Lagrange's interpolation formula is used to reconstruct the polynomial $F(x)$ from any k or more of the shares. Blakley (1979) proposed another threshold scheme based on plane geometry to share a secret in the same year. Asmuth and Bloom (1983) and Mignotte (1983) proposed new threshold schemes based on the Chinese Remainder Theorem (CRT) in 1983.

Threshold based schemes are widely used to share secret images. Blakley's sharing scheme is used to share a secret image among many participants (Chen and Fu, 2008; Tso, 2008). Number theory based secret sharing approaches are also used (Chen and Chang, 2007; Shyu and Chen, 2008). But most of the recent secret image sharing is based on Shamir's scheme.

Thien and Lin (2002) use Shamir's approach to share a secret image among n participants. Then, Lin and Tsai (2004) used steganography to hide the share images into natural looking cover images. Their method is improved as they used a parity bit to avoid fake stego blocks. After this work, Yang et al. (2007) underlined some drawbacks in Lin and Tsai's work and used the Galois Field $GF(2^8)$ to provide distortion free secret image sharing. Their method can detect 50% of fake stego blocks. Chang et al. (2008), provides

* Corresponding author. Tel.: +90 533 2277990.
E-mail addresses: guzin@ieee.org (G. Ulutas), ulutas@ieee.org (M. Ulutas), vasif@ktu.edu.tr (V.V. Nabiyev).

0167-8655/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.patrec.2012.10.017

better authentication using the CRT. Four authentication bits are used to detect fake stego blocks. Their method can detect 93% of the fake stego blocks. However, the visual quality of stego images is worse compared to other works reported in the literature (Yang and Ciou, 2009). Eslami and Ahmadabadi (2011) proposed a new embedding method to use all the capacity of cover images for data hiding. Block size of their method is determined dynamically depending on the size of secret data. Their work also used a new authentication with four bits. Authentication bits of a hidden data block depend on both current block and previous blocks' authentication bits. Therefore, their method can detect a fake stego block with probability 15/16 $\cong$ 0.93. However, their method has a drawback due to chaining of authentication bits. If a block in the stego image is modified, remaining stego blocks cannot be verified by the method even if they are authentic. Thus, the method cannot reconstruct corresponding secret pixel values. Another drawback of their method is the constant number of bits used for authentication. Four bits string is used for authentication regardless of block size.

Adaptive authentication is proposed in this work allowing increased number of bits for larger blocks improving authentication strength. Shared values are embedded into blocks using 8-ary exploiting modification direction (EMD) method (Lee et al., 2007). Size of the block controls authentication strength of the sharing. A four-bit string is used for authentication of small blocks and more than four bits can be used for larger blocks. The method verifies current block with authentication string embedded in the next block. Thus, an altered stego block only affects the previous block, not all the remaining blocks as in Eslami and Ahmadabadi (2011) work. For example, second and third stego blocks are assumed modified if only third block fails authentication in a stego image. In other words, all blocks after an altered block can be authenticated by the proposed method. On the contrary, Eslami and Ahmadabadi (2011) work assumes all blocks after an altered block as modified. Our proposed method also generates better visual quality stego images with peak to signal noise ratio (PSNR) over 48 dB on the average for block sizes greater than 8. It is not easy for the human visual system to perceive such a small distortion in an image.

The rest of this paper is organizes as follows: 8-ary EMD method is explained in the following section. Third section explains the details of the proposed method. Experimental results are provided in Section 4. Finally, conclusions are drawn in Section 5.

2. Exploiting modification direction

Zhang and Wang (2006) proposed a steganographic method to exploit the modification directions for hiding secret data. Their method assumes that each secret digit in a $(2n + 1)$-ary notational system is carried by n cover pixels and is denoted by $(g_1,\ g_2, \ldots, g_n)$. There are $2n$ possible ways of modification if only one pixel in a group of n pixels is incremented or decremented by one. No modification is another possibility which increases the total number of modifications to $(2n + 1)$. Lee et al. (2007) proposed a method based on EMD. Their work revises the EMD extraction function to improve the embedding rate. Let the embedding rate, the number of secret bits embedded per cover pixel be R. A secret digit in the $(2n + 1)$-ary notational system, representing $\log_2(2n + 1)$ bits, is embedded into n pixels, among which one pixel is incremented or decremented by 1 with probability $2n/(2n + 1)$. Thus, the embedding rate R is defined as

$$R = \frac{\log_2(2n+1)}{n}. \tag{1}$$

The embedding rate decreases as n is increased. The EMD method has maximum embedding capacity for 5-ary notational system where each secret digit is carried out by two cover pixels. However, the method has a lower embedding capacity for larger values of n. Lee et al. (2007) proposed a method to increase the embedding rate of the EMD method. Their embedding rate is 1.5 times that of the EMD embedding for equal stego image quality.

Their method converts secret image pixels to a sequence of secret digits in 8-ary notational system where each digit can be represented by three bits. Cover image pixels are grouped into two pixel sequences (g_1, g_2). A cover pixel group can be used to carry one secret digit in 8-ary notational system. A secret digit is coded into corresponding cover pixel block by the extraction function given below:

$$f(g_1, g_2) = (g_1 \cdot 1 + g_2 \cdot 3) \bmod 8 \tag{2}$$

The extraction function used by their method is different from Zhang et al.'s extraction function. Only one pixel is incremented or decremented by one if the value of extraction algorithm is not equal to the secret digit. Lee et al.'s method is used to hide the shared values into cover pixels in the proposed method as outlined below. But the proposed method also necessitates to hide the shared values in [0–15] range. So, extraction function used in their work is modified as $f(g_1,\ g_2) = (g_1 \cdot 1 + g_2 \cdot 3) \bmod 16$ to hide any secret digit in this range by modifying one cover pixel or both of them by increasing or decreasing their values by one or two. Let the result of extraction function for the current cover pixel group be 10 and secret digit be 8. First pixel is decremented by two to hide the secret digit. If secret digit is 6, both of the cover pixels are decremented by one. If secret digit is 2, both of the cover pixels are decremented by two. Assume that the result of extraction function be v. All the digits in $[v - 8, v + 8]$ range can be generated by decrementing or incrementing the cover pixel values by one or two. This range corresponds to the digits in base 16.

3. Proposed method

The proposed method consists of a sharing algorithm where modified EMD is used to embed shares generated by Shamir's polynomial and a retrieving algorithm where authenticated shares are used to reconstruct the secret by Lagrange's interpolation. The details of the sharing algorithm which is given in Fig. 1 as a flowchart, and retrieving algorithm are explained in this section.

3.1. Sharing algorithm

The algorithm shares the secret image among n participants and embeds shares into n cover images. Secret image is divided into sections of k pixels to construct Shamir's polynomial $F(x)$ of degree $k - 1$. Share computation is performed by evaluating the polynomial for unique x values, x_i, $i = 1, \ldots, n$ corresponding to participants. Shared values and authentication bits of both current and previous block are then embedded into corresponding cover blocks. Authentication provides a mechanism to detect any alteration on the stego image blocks.

Let a grey level secret image be $S = \{s_{ij} | s_{ij} \in [0 - 255],\ 1 \leqslant i \leqslant S_N,\ 1 \leqslant j \leqslant S_M\}$.. The secret image is split into k pixel sections to construct $k - 1$ degree polynomials. The first $k - 1$ degree polynomial constructed from sections $(s_1, s_2, \ldots, s_k)$ is given below.

$$F(x) = \left(s_1 + s_2 x + s_3 x^2 + \cdots + s_k x^{k-1}\right) \bmod 257 \tag{3}$$

The prime modulus used in Shamir's polynomial is selected as 257 since it is the first prime number that covers the range of an 8-bit grey level secret image. Shares $F(x_1), F(x_2), \ldots, F(x_n)$ are computed by evaluating the polynomial for x_i corresponding to ith participant. For a shared value of 256, one pixel of the current block is increased or decreased by one and new shared values are calculated again to create shared values in [0–255] range. Then shares

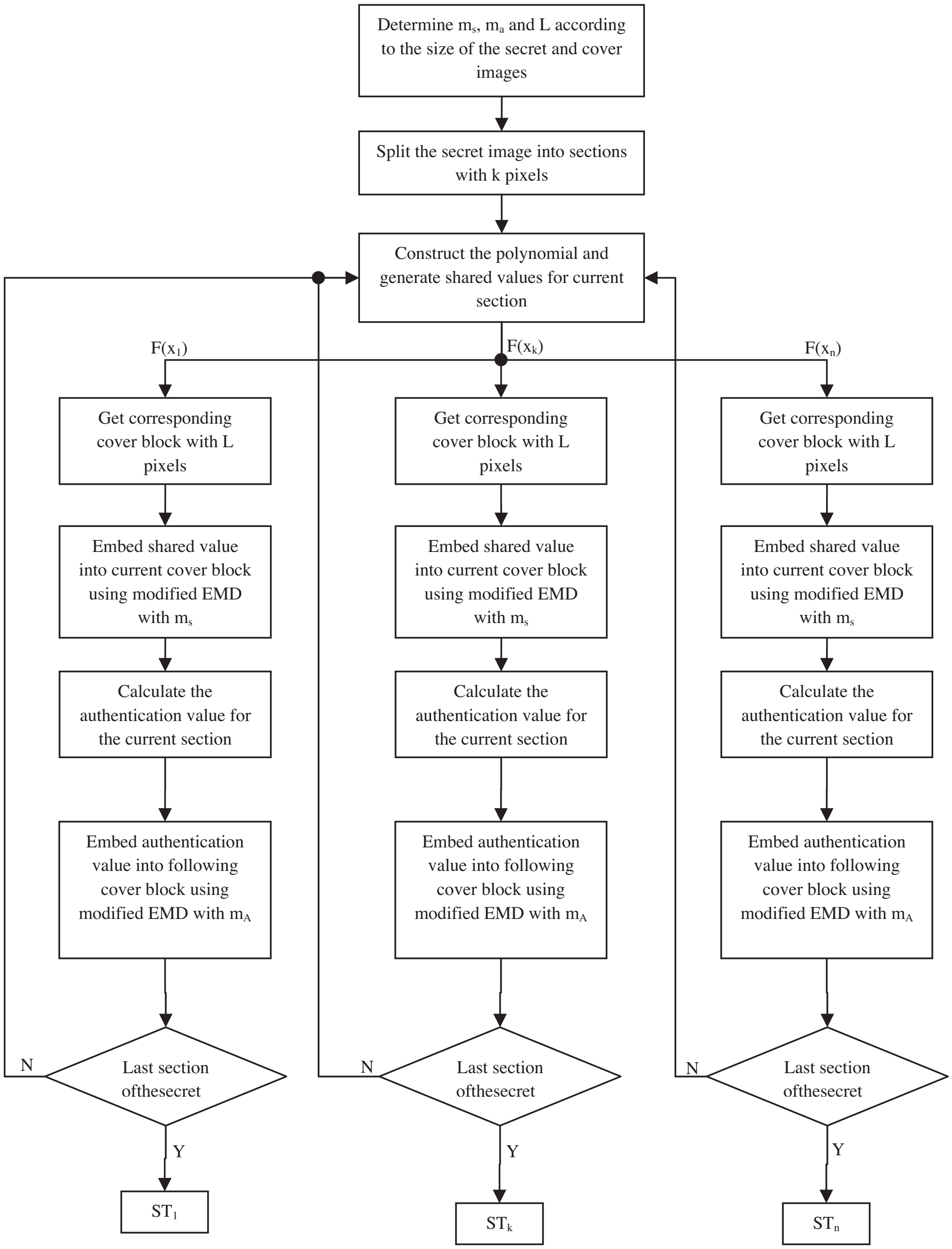

Determine m_s, m_a and L according to the size of the secret and cover images
Split the secret image into sections with k pixels
Construct the polynomial and generate shared values for current section
F(x_1)
F(x_k)
F(x_n)
Get corresponding cover block with L pixels
Embed shared value into current cover block using modified EMD with m_s
Calculate the authentication value for the current section
Embed authentication value into following cover block using modified EMD with m_A
Last section ofthesecret
N
Y
ST_1
ST_k
ST_n

Fig. 1. Flow chart of the sharing algorithm.

are embedded into corresponding cover blocks of size L defined in (4) in terms of secret image size, cover image size and the threshold value:

$$L = \lfloor (C_N \cdot C_M \cdot k)/(S_N \cdot S_M) \rfloor \tag{4}$$

The cover image is split into size L blocks and blocks in the kth cover image C^k are $C_1^k, C_2^k, \ldots, C^k_{\lfloor C_N C_M / L \rfloor}$. Shared values are converted into base m_s notation depending on the value of L represented with $\lceil \log_{m_s} 255 \rceil$ digits $(sd_1, \ldots, sd_{\lceil \log_{m_S} 255 \rceil})_{m_S}$. Smallest value of m_S which satisfies the following condition should be selected:

$$L - 2 \geqslant 2\left\lceil \log_{m_s} 255 \right\rceil, \quad m_S \in \{8,\ 16\} \tag{5}$$

One shared digit of a shared value is embedded into corresponding two consecutive pixels of the current cover block C_i^k using modified EMD method proposed in (Lee et al., 2007). Let pixels of C_i^k are $c_{i1}^k, c_{i2}^k, \ldots, c_{iL}^k$. The extraction function used for hiding a shared digit into corresponding first two pixels of ith group at kth cover image (c_{i1}^k, c_{i2}^k) is given in (6). Alteration interval $[-a, a]$ of cover pixels is also determined by the value of m_s as in (6).

$$\begin{aligned} f_e &= (c_{i1}^k + 3c_{i2}^k) \bmod m_s \\ m_s &= 8 \Rightarrow a = 1 \\ m_s &= 16 \Rightarrow a = 2 \end{aligned} \tag{6}$$

Sharing algorithm evaluates the extraction function for the first two pixels of the ith cover block, $f_e = (c_{i1}^k + 3c_{i2}^k) \bmod m_s$. These pixels are modified accordingly in [−a, a] range to yield a value equal to the first shared digit sd_1 unless f_e is equal to sd_1. Corresponding stego pixels are obtained after the alteration. Other shared digits (sd_2, sd_3) are also embedded into two consecutive pair of the cover pixels $(c_{i3}^k c_{i4}^k)$, $(c_{i5}^k c_{i6}^k)$ likewise.

Current cover block is modified to accommodate the shared digits as explained above. Most of the techniques reported in the literature use simple LSB embedding technique to hide the shared value. After embedding the shared value, authentication value of the current block is calculated and embedded into bit positions determined by the researchers. The proposed method uses modified EMD method to hide the shared digits into cover pixels of the current block. Such an alteration makes it impossible to estimate how many bits of the cover pixels are modified during the embedding procedure. The proposed method does not predict bit positions of the authentication values for the current block. So, authentication value of the current block is calculated after hiding the shared digits on the current block and embedded into following cover block. In other words, each cover block holds authentication information of the previous block.

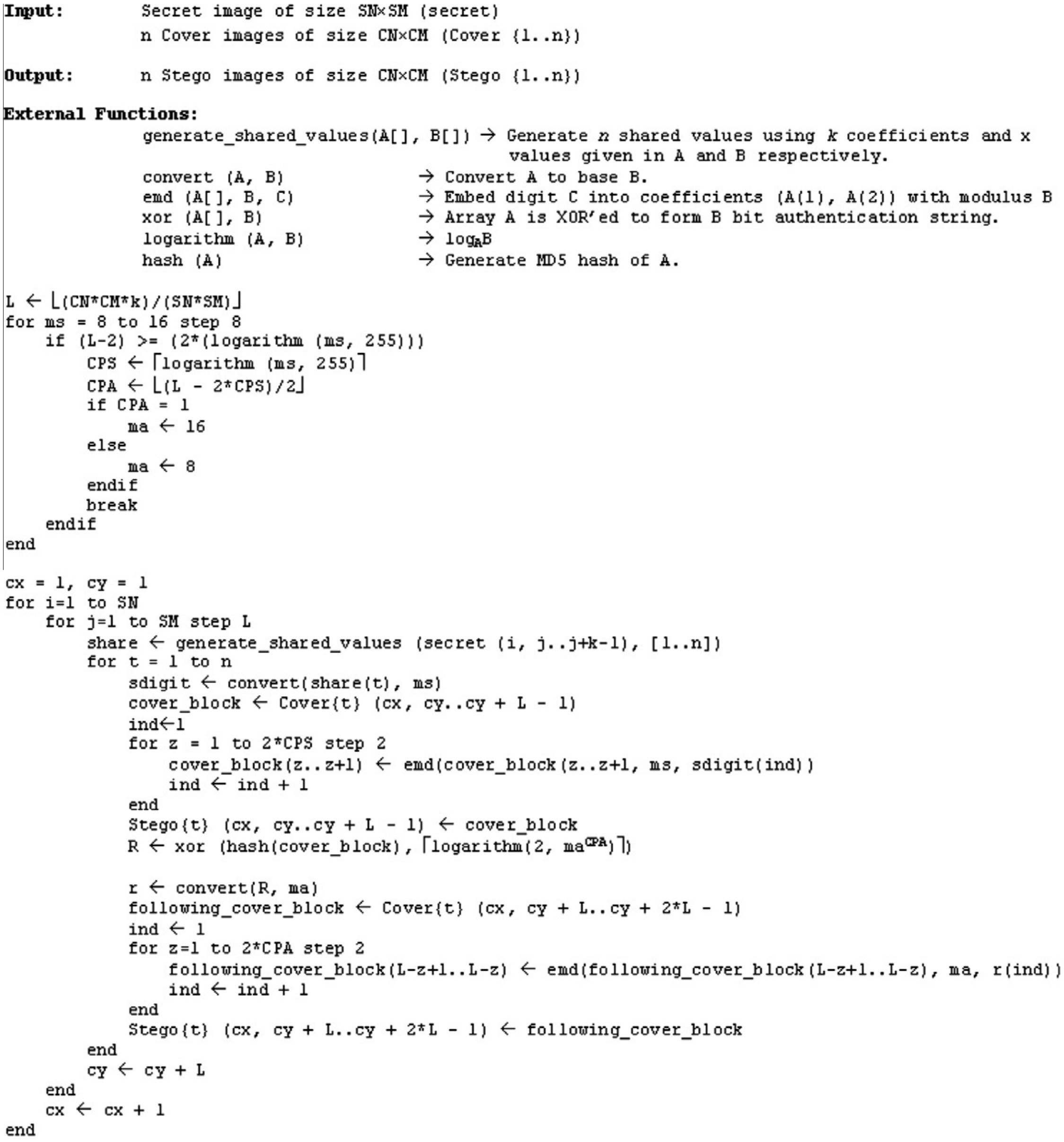
The proposed method uses $CP_S = \left\lceil \log_{m_S} 255 \right\rceil$ pixel pairs of the cover block to embed shared values and the rest of the pixels are used to encode authentication value. Assume that CP_A is the number of cover pixel pairs used to hide the authentication value. The value of CP_A is determined using the values of L and CP_S as in (7). CP_A also determines the base m_A used to convert the authentication value:

$$\begin{aligned} CP_A &= \lfloor (L - 2 \cdot CP_S)/2 \rfloor = 1 \Rightarrow m_A = 16 \\ CP_A &= \lfloor (L - 2 \cdot CP_S)/2 \rfloor \geqslant 2 \Rightarrow m_A = 8 \end{aligned} \tag{7}$$

Sharing algorithm calculates the authentication value of the current stego block according to the value of CP_A and m_A. Let ith stego block of the kth stego image denoted by ST_i^k. A keyed hash function, $H(\cdot)$, is computed on the current block. Result of the hash function is XOR'ed according to the number of authentication bits b as in (8).

$$R = \left\langle H\left(ST_i^k\right) \right\rangle_b, \quad b = \left\lceil \log_2 m_A^{CP_A} \right\rceil \tag{8}$$

Authentication value of the current stego block R is embedded into the following cover block. Authentication value R is converted to authentication digits in base m_A notation as $r_1, \ldots, r_{CP_A}$. The last $2 \cdot CP_A$ bytes of the following cover block $\left(c^k_{i(j+2L-2CP_A)}, \ldots, c^k_{i(j+2L-1)}\right)$ are used to hide the authentication digits $(r_1, \ldots, r_{CP_A})$. Modified



```
Input:        Secret image of size SN×SM (secret)
              n Cover images of size CN×CM (Cover {1..n})

Output:       n Stego images of size CN×CM (Stego {1..n})

External Functions:
              generate_shared_values(A[], B[]) → Generate n shared values using k coefficients and x
                                                 values given in A and B respectively.
              convert (A, B)              → Convert A to base B.
              emd (A[], B, C)             → Embed digit C into coefficients (A(1), A(2)) with modulus B
              xor (A[], B)                → Array A is XOR'ed to form B bit authentication string.
              logarithm (A, B)            → log_AB
              hash (A)                    → Generate MD5 hash of A.

L ← ⌊(CN*CM*k)/(SN*SM)⌋
for ms = 8 to 16 step 8
    if (L-2) >= (2*(logarithm (ms, 255)))
        CPS ← ⌈logarithm (ms, 255)⌉
        CPA ← ⌊(L - 2*CPS)/2⌋
        if CPA = 1
            ma ← 16
        else
            ma ← 8
        endif
        break
    endif
end

cx = 1, cy = 1
for i=1 to SN
    for j=1 to SM step L
        share ← generate_shared_values (secret (i, j..j+k-1), [1..n])
        for t = 1 to n
            sdigit ← convert(share(t), ms)
            cover_block ← Cover{t} (cx, cy..cy + L - 1)
            ind←1
            for z = 1 to 2*CPS step 2
                cover_block(z..z+1) ← emd(cover_block(z..z+1, ms, sdigit(ind))
                ind ← ind + 1
            end
            Stego{t} (cx, cy..cy + L - 1) ← cover_block
            R ← xor (hash(cover_block), ⌈logarithm(2, ma^CPA)⌉)

            r ← convert(R, ma)
            following_cover_block ← Cover{t} (cx, cy + L..cy + 2*L - 1)
            ind ← 1
            for z=1 to 2*CPA step 2
                following_cover_block(L-z+1..L-z) ← emd(following_cover_block(L-z+1..L-z), ma, r(ind))
                ind ← ind + 1
            end
            Stego{t} (cx, cy + L..cy + 2*L - 1) ← following_cover_block
        end
        cy ← cy + L
    end
    cx ← cx + 1
end
```

Fig. 2. Pseudo code of the sharing algorithm.

EMD method is also used during the embedding of the authentication digits. Each authentication digit is embedded into corresponding two consecutive pixels of the cover block. Assume that current cover pixels used for hiding an authentication digit be $\left(c^k_{i(j+2L-2)}, c^k_{i(j+2L-1)}\right)$. The modulus value used in the extraction function given in (9) to code the authentication digit is m_A instead of m_S:

$$\begin{aligned} f_e &= \left(c^k_{i(j+2L-2)} + 3c^k_{i(j+2L-1)}\right) \bmod m_A \\ m_A &= 8 \Rightarrow a = 1 \\ m_A &= 16 \Rightarrow a = 2 \end{aligned} \tag{9}$$

The proposed method embeds the authentication value of the current cover block into the next cover block. During the retrieving algorithm, a stego block should be authenticated using the authentication value embedded in the next stego block. Such an authentication mechanism provides the method improved authentication strength. Eslami and Ahmadabadi (2011) method authenticates a block with reconstructed shared value from the previous block. Thus, they claim that the probability of authenticating a modified block is reduced to 1/4 · 1/4 = 1/16 due to the conditional probability. Even though their method has improved authentication strength, it has a drawback. Shared value cannot be revealed for a block if previous block is modified, causing a domino effect on the rest of the blocks since the retrieving algorithm necessitates authentication of both previous and current blocks to successfully process the blocks to reveal the secret. Thus, all blocks following a modified block of the stego image can neither be authenticated nor used to reconstruct the secret image by the retrieving algorithm. In other words, modification of any stego block causes rejection of all remaining blocks. On the other hand, the proposed method authenticates blocks individually and rejects only modified blocks, which limits the propagation of error. Pseudo code of the sharing algorithm is given in Fig. 2.

3.2. Retrieving algorithm

Retrieving algorithm should gather at least k stego images to reconstruct the original secret image. Stego images are divided into blocks. Authentication value is calculated for the current stego block and compared with the embedded authentication value that is extracted from the next stego block. Stego blocks are verified if the calculated authentication value and extracted authentication values are equal. Otherwise, corresponding k secret pixels in the recovered secret image is marked as modified. Shared values embedded in stego blocks are extracted and used with Lagrange's interpolation to recover the corresponding secret pixel values upon successful verification of k stego block is verified.

Let k stego images denoted by $ST^1, ST^2, \ldots, ST^k$. Stego images are divided into blocks of size L during the retrieving algorithm. $\lfloor NM/L \rfloor$ blocks are created for a stego image of size $N \times M$. Blocks of the kth stego image be denoted by $ST^k_1, ST^k_2, \ldots, ST^k_{\lfloor NM/L \rfloor}$. Each stego block has L pixels. Retrieving algorithm calculates the authentication value on the current block to verify the stego block. The original authentication value is embedded into the following cover block during the sharing algorithm. Last CP_A pixels of the following stego block accommodate the authentication value about the current stego block. Let pixels of (i – 1)th block in kth stego image be denoted by $st^k_{i-11} \ldots st^k_{i-1L}$. Embedded authentication digits during the sharing algorithm for (i – 1)th block are extracted from the ith block using (10):

$$a^k_{(i-1)t} = \left(st^k_{i(L-2CP_A+2(t-1)+1)} + 3 \cdot st^k_{i(L-2CP_A+2t)}\right) \bmod m_A \quad t = 1 \ldots CP_A \tag{10}$$

Digits of the embedded authentication value in base m_A for (i – 1)th cover block are represented by $a^k_{(i-1)1} a^k_{(i-1)2} \cdots a^k_{(i-1)CP_A}$. Its decimal value is calculated using (11):

$$A_{i-1} = a^k_{(i-1)1} \cdot m_A^{(CP_A-1)} + \cdots + a^k_{(i-1)CP_A} \tag{11}$$

The dealer also calculates the current authentication value for (i – 1)th cover block using (12):

$$R_{i-1} = \left\langle H\left(ST^k_{i-1}\right)\right\rangle_b, \quad b = \left\lceil \log_2 m_A^{CP_A} \right\rceil \tag{12}$$

Current cover block is verified if the calculated authentication value R_{i-1} and extracted authentication value A_{i-1} are equal. Otherwise corresponding k pixels in the reconstructed secret image is marked as modified. Shared value is extracted if current block is verified successfully. Shared digits in base m_s embedded in (i – 1)th stego block of kth stego image are extracted using (13):

$$sd^k_{(i-1)t} = \left(st^k_{(i-1)(2(t-1)+1)} + 3 \cdot st^k_{(i-1)(2t)}\right) \bmod m_S, \quad t = 1 \cdots CP_S \tag{13}$$

Shared digits in base m_S are converted to decimal notation using (14):

$$F(x_k) = sd^k_{(i-1)1} \cdot m_S^{(CP_S-1)} + \cdots + sd^k_{(i-1)t} \tag{14}$$

Shared values $F(x_1), \ldots, F(x_{k-1})$ are obtained from other $k-1$ stego blocks in the same manner. Lagrange's interpolation technique is then used to reconstruct corresponding k secret pixel values from shared values upon successful verification of all cover blocks. Reconstructed secret pixel values are marked as modified corresponding to blocks that cannot be verified. The steps of the retrieving algorithm are given below:

1. k Stego images are gathered to reconstruct the secret image $ST^1, ST^2, \ldots, ST^k$.
2. Stego images are divided into blocks of size L.
3. Repeat the following steps for $1 \leqslant i \leqslant \lfloor NM/L \rfloor$.
   - 3.1. Obtain corresponding ith stego blocks in k stego images. $ST^1_i, ST^2_i, \ldots, ST^k_i$.
   - 3.2. Repeat the following steps for $1 \leqslant j \leqslant k$.
     - 3.2.1. Extract the embedded authentication value from the following stego block ST^j_{i+1} using (10) and (11).
     - 3.2.2. Calculate current authentication value for ST^j_i using (12).
     - 3.2.3. Go to step 3.2.5 if extracted authentication value and current authentication values are equal.
     - 3.2.4. Mark the corresponding k pixels in the reconstructed secret image as modified and return to step 3.
     - 3.2.5. Extract the shared value $F(x_j)$ using (13) and (14).
   - 3.3. Use Lagrange's interpolation technique to reconstruct Shamir's polynomial from the shared values. k coefficients of the reconstructed polynomial are the corresponding secret pixel values in the reconstructed secret image.

4. Experimental results

The effectiveness of the proposed method is tested by experiments and the results are summarized in this section. There are two metrics used in the literature to indicate the performance of the secret image sharing approaches: PSNR of the stego images and detection ratio of fake stego blocks. In this regard, a (2,3) secret sharing scheme is used for the first experiment. Secret is selected to be gray level "Jet" image of size 256 × 256 as shown in Fig. 3(a). Three gray level images "Lena", "Lighthouse" and "Parrots" of size 512 × 512 given in Fig. 3(b)–(d) are selected as cover images among the well known test images. The proposed method shares the secret image among the three participants using the sharing algorithm. Stego images given in Fig. 3(e)–(g) are cover images modified by the EMD to accommodate shared values. Human visual system (HVS) can not perceive the modification if the absolute value of the difference between the cover and stego images is small enough. Even though we cannot perceive the modification, peak to signal noise ratio (PSNR) is generally used as a quantitative measure to indicate the amount of modification. PSNR is defined as

$$PSNR = 10 \times \log_{10} \frac{(255)^2}{MSE} dB \tag{15}$$

where *MSE* is the mean squared error between the cover image and stego image and 255 is the maximum intensity value used in 8-bit monochrome images. *MSE* for a cover image of $C_N \times C_M$ is defined as

$$MSE = \frac{1}{C_N C_M} \sum_{i=1}^{C_N} \sum_{j=1}^{C_M} (C_{ij} - ST_{ij})^2 \tag{16}$$

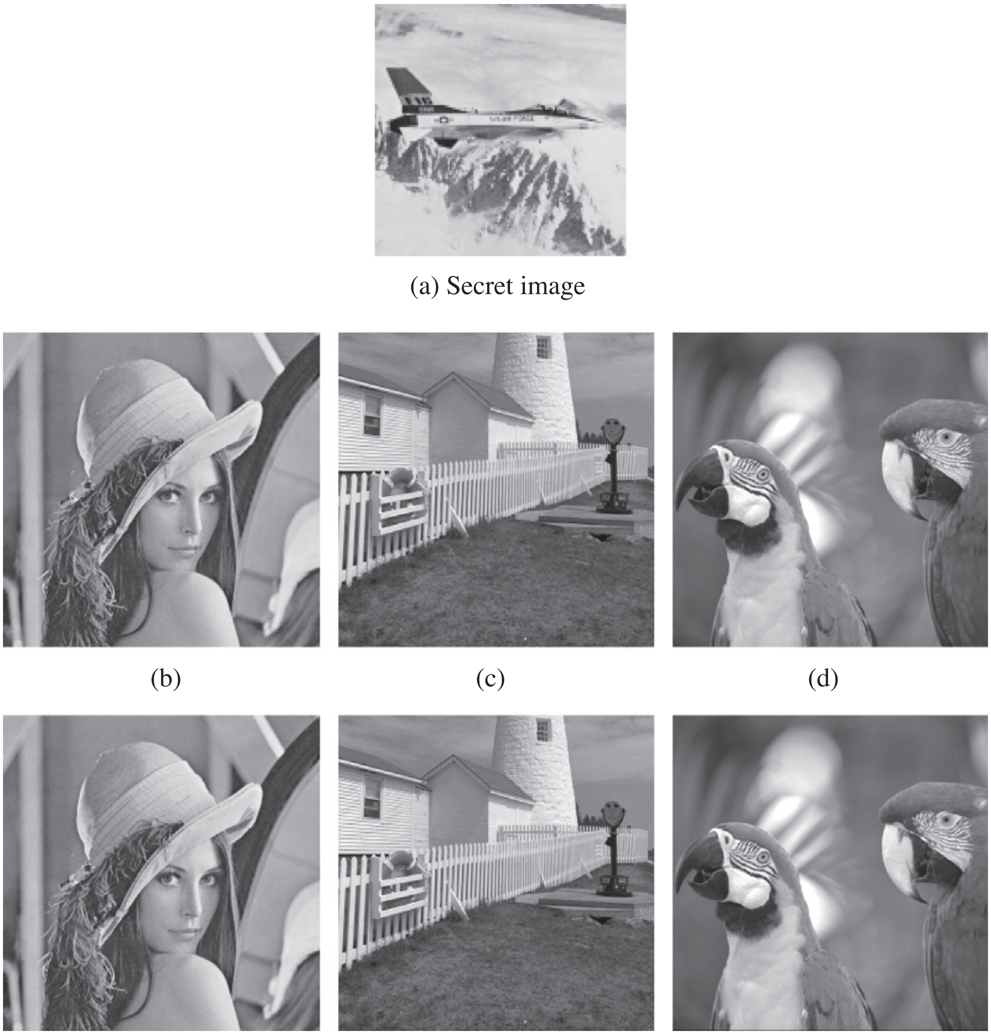
PSNR of stego images generated by the proposed method are approximately 48.6 dB as indicated in Fig. 3. Alterations on the stego images cannot be perceived by the HVS. Block size is determined to be 8 for this experiment by using (4) and 4 authentication bits are used to verify stego blocks in this experiment due to the size of the block.

The PSNR of stego images for different block sizes is computed during the second experiment. Secret image of size 256 × 256 used in the first experiment is also used in the second experiment. Cover images used in the first experiment are resized to 362 × 362 pixels to test performance of the scheme for different block sizes. Block sizes generated by both methods for $k = 3$, 4, 5, 6 and 7 are 6, 8, 10, 12 and 14 respectively. As a result, Table 1 illustrates PSNR as a function of block size and threshold value *k*. The table clearly indicates that visual quality of stego images improve as the block size is increased. 48 dB PSNR or better is computed for block size of greater than 8. Secret capacity of both methods is equal because of the method used in the sharing algorithm to determine the block size.

Table 1
Comparision of the two methods according to authentication capability and PSNR.

Proposed method				Eslami and Ahmadabadi (2011)			
(*k*)/ Block size	Number of auth. bits	Detection ratio(DR)	PSNR (dB)	(*k*)/ Block size	Number of auth. bits	Detection ratio(DR)	PSNR (dB)
3/6	4	0.94	45.95	3/6	4	–	45.60
4/8	4	0.94	48.39	4/8	4	–	48.03
5/10	6	0.98	49.90	5/10	4	–	50.69
6/12	9	0.998	49.91	6/12	4	–	51.47
7/14	12	0.999	50.01	7/14	4	–	52.10

–: DR values of Eslami and Ahmadabadi (2011) work exceeds upper limit of DR.



(a) Secret image

(b)

(c)

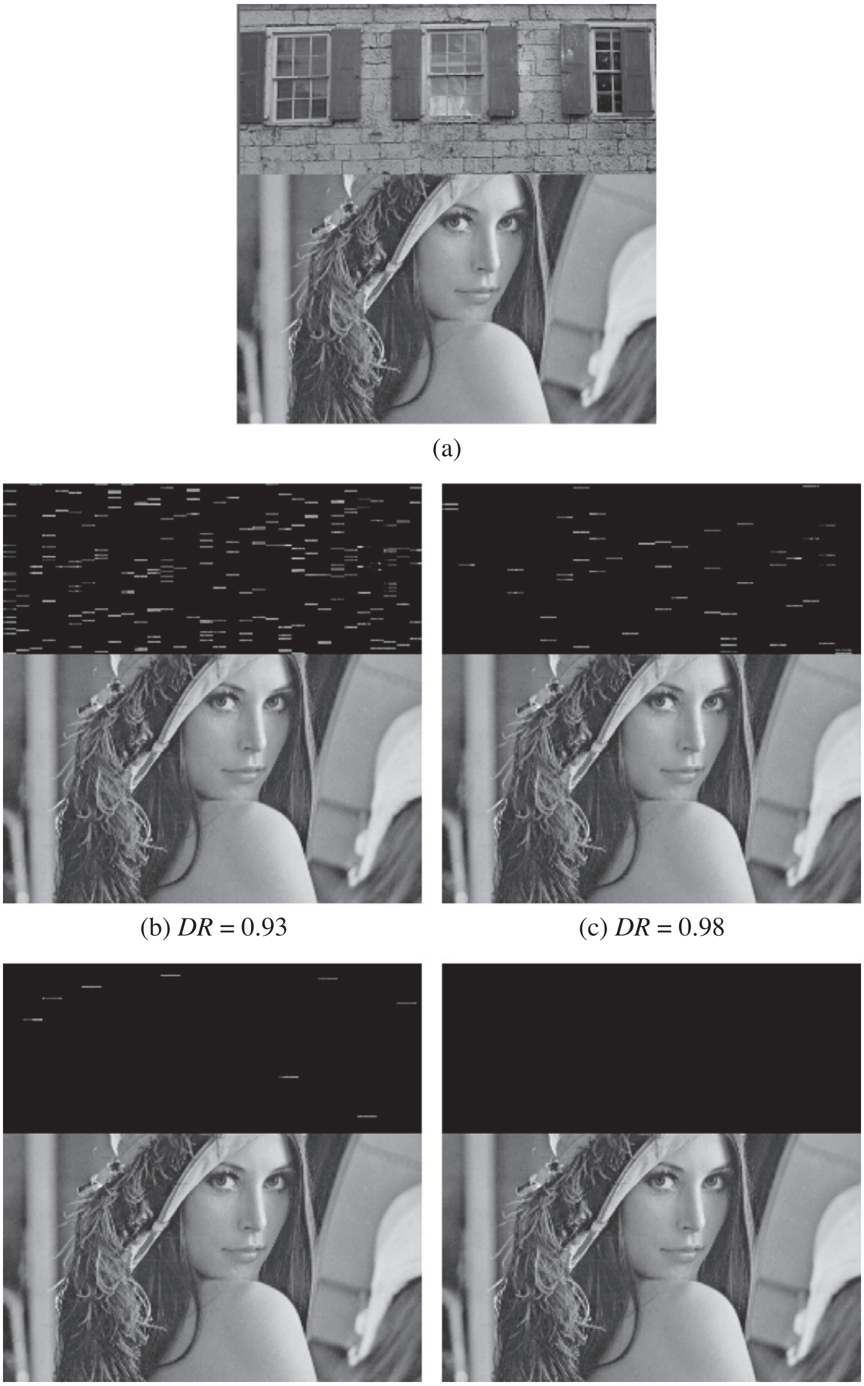
(d)

(e) 48.45 dB

(f) 48.62 dB

(g) 48.87 dB

Fig. 3. (a) Secret image of size 256 × 256 (b)–(d) Cover images of size 512 × 512 (e)–(g) Stego images with PSNR.



(a)

(b) $DR = 0.93$

(c) $DR = 0.98$

(d) $DR = 0.99$

(e) $DR = 1.0$

Fig. 4. (a) Stego image with fake region. (b)–(d) Detection ratio of the proposed method when 4, 6, 9, 12 authentication bits are used respectively.

Authentication strength of the proposed method is also tested during the experiments. A 208 × 512 fake region is inserted into the upper side of the stego image Lena created in the first experiment as shown in Fig. 4(a). The retrieving algorithm must verify the stego images before reconstruction of the secret pixels. In order to test the effectiveness of the authentication procedure, Chang et al.'s detection ratio (*DR*) metric defined in 2008 is used. DR is defined as the ratio of number of tampered pixels detected (*NTPD*) to number of tampered pixels (*NTP*), $DR = NTPD/NTP$. *DR* is in [0–1] range and 0 stands for a method which cannot detect any fake stego pixels. The authentication strength of the proposed method for block sizes of 8, 10, 12 and 14 are tested in this experiment. Block size also determines the number of authentication bits as defined in (7). The method uses 4, 6, 9 and 12 bit authentication for corresponding block sizes respectively. *DR* of authentication procedure for different block sizes is given in Fig. 4(b)–(e) to illustrate the effectiveness. Fake regions are detected with probability 0.93 when four authentication bits are used. Detection ratio of the proposed method improves as the block size is increased. Detection ratio of the method with 9 bit authentication is approximately 0.99. Fake region is detected when the number of authentication bits is 12 as shown in Fig. 4(e). Proposed method provides flexible

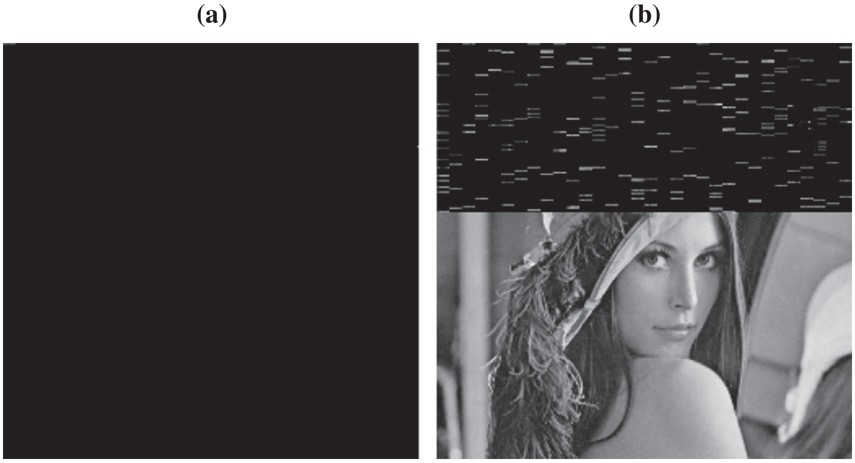


Fig. 5. Comparision of the authentication capability of the two methods when the number of authentication bits is 4 (a) Eslami and Ahmadabadi (2011) method (b) proposed method.

authentication strength to the secret image sharing scheme depending on the size of the blocks as indicated in this experiment.

Eslami and Ahmadabadi (2011) reports in their work that four bits are used for authentication regardless of the block size. However, the most important drawback of their method is chaining in the authentication process. Their method necessitates the shared value obtained in the previous block to authenticate current block. In other words, the current block cannot be authenticated if the previous block is modified. That means, any disruption breaks the authentication process at that point even if the rest of the blocks are authentic. Their work does not report DR value for demonstrating authentication capability of their method. But, DR value defined in Chang et al.'s work is used as a metric to test the authentication capability of the secret sharing schemes in the literature.

A $(2,3)$ secret sharing scheme is used to compare the authentication capability of the proposed method and Eslami and Ahmadabadi (2011) work. Secret image given in Fig. 3(a) and cover images given in Fig. 3(b)–(d) are used in this experiment. A fake stego image given in Fig. 4(a) is created to test the authentication capability of both the proposed and Eslami and Ahmadabadi (2011) works.

The result of the verification procedure of the proposed method and Eslami and Ahmadabadi (2011) work is shown in Fig. 5 for block size of 8. Eslami and Ahmadabadi (2011) work considers the whole stego image as corrupt even if only upper side of it is corrupt. It assumes authentic stego blocks as corrupt as can be seen in Fig. 5(a). Beside, *DR* value for Eslami and Ahmadabadi (2011) work is calculated as $512 \times 512/208 \times 512 \cong 2.4$ even though *DR* cannot exceed 1. Their method detects 512×512 stego pixels as fake although fake stego image size is 208×512 pixels. The proposed method detects fake stego blocks with probability 0.93 as can be seen in Fig. 5(b). Authentic stego blocks are verified successfully and corresponding pixels in the reconstructed secret image is retrieved by the proposed method. But Eslami and Ahmadabadi (2011) work assumes whole stego image as fake even if one pixel of it is modified due to the chaining drawback in their work.

Table 1 compares detection ratio and block size of both methods. The proposed method exhibits improved authentication as the block size is increased. The proposed method can use up to 12 bit for authentication with 50 dB PSNR for block size of 14. Dynamic authentication strength of the proposed method is desirable due to the user defined trade-off between authentication and image quality. However, DR values of Eslami and Ahmadabadi (2011) work cannot be given in this table due to the drawback in their method as reported in the previous experiment. DR values of their method exceed the upper limit of DR. Their work also does not report any DR value.

Dynamic authentication capability depending on the size of the secret and cover image and the value of *k* is provided in this work. Authentication capability of the proposed method improves as the block size increase contrary to Eslami and Ahmadabadi (2011) where authentication fails due to the chaining mechanism used in their algorithm. Stego blocks following an accidentally corrupted block during transmission are assumed to be corrupt because of the chaining mechanism. However, proposed method has a downgrade on the visual quality compared to their method for increasing value of *k*. The proposed method has dynamic authentication capability and it is not affected by the problem that exists in Eslami and Ahmadabadi (2011) work.

5. Conclusion

Recently, Eslami and Ahmadabadi (2011) proposed a secret sharing method that uses all pixels of the cover image for embedding and block authentication chaining to increase the number of authentication bits. Conditional verification because of chaining causes authentication failure for the rest of the stego blocks even if a single block is modified unintentionally and the secret cannot be reconstructed. On the other hand, their method uses four bits of the block for all block sizes. Authentication strength depending on size of blocks is proposed in this work. At least four bits is used for authentication but it can be increased for larger blocks. Larger blocks can be used to improve authentication strength of the proposed method. Block contents are assumed valid if the next block carrying the authentication bits of the former block is authenticated. Thus, transmission errors causing an authentic block to look like a fake block affects only the former block, unlike the case when using the approach of Eslami and Ahmadabadi (2011) Also, stego images produced by the proposed method have better visual quality and yields over 48 dB PSNR which cannot be perceived by the HVS for block size larger than 8.

References

Asmuth, C., Bloom, J., 1983. A modular approach to key safeguarding. Trans. Inform. Theory 29 (2), 208–210.

Blakley, G.R., 1979. Safeguarding cryptographic keys. In: Proc. National Computer Conf., pp. 313–317.

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

Chen, C.-C., Chang, C.-C., 2007. Secret image sharing using quadratic residues. In: Intelligent Information Hiding and Multimedia, Signal Processing, pp. 515–518.

Chen, C.-C., Fu, W.-Y., 2008. A geometry based secret image sharing approach. J. Inform. Sci. Eng. 24 (5), 1567–1577.

Eslami, Z., Ahmadabadi, J.Z., 2011. Secret image sharing with authentication-chaining and dynamic embedding. J. Systems Software 84 (5), 803–809.

Lee, C.-F., Wang, Y.-R., Chang, C.-C., 2007. A steganographic method with high embedding capacity by improving exploiting modification direction. In: Intelligent Information Hiding and Multimedia, Signal Processing, Koahsiung, pp. 497– 500.

Lin, C.-C., Tsai, W.-H., 2004. Secret image sharing with steganography and authentication. J. Systems Software 73 (3), 405–414.

Mignotte, M., 1983. How to share a secret. Lect. Notes Comput. Sci. 149, 371–375.

Shamir, A., 1979. How to share a secret. Comm. ACM 22 (11), 612–613.

Shyu, S.J., Chen, Y.-R., 2008. Threshold secret image sharing by Chinese remainder theorem. In: IEEE APSCC, pp. 1332–1337.

Thien, C.-C., Lin, J.-C., 2002. Secret image sharing. Comput. Graphics 26, 765–770.

Tso, H.-K., 2008. Sharing secret images using Blakley's concept. Opt. Eng. 47 (7), 1–3.

Yang, C.-N., Chen, T.-S., Yu, K.H., Wang, C.-C., 2007. Improvements of image sharing with steganography and authentication. J. Systems Software 80 (7), 1070–1076.

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

Zhang, X., Wang, S., 2006. Efficient steganographic embedding by exploiting modification direction. IEEE Comm. Lett. 10 (11), 781–783.