

## Research Article

# A Secret Image Sharing Method Using Integer Wavelet Transform

Chin-Pan Huang<sup>1</sup> and Ching-Chung Li<sup>2</sup>

<sup>1</sup>Department of Computer and Communication Engineering, Ming Chuan University, Taoyuan 333, Taiwan

<sup>2</sup>Department of Electrical and Computer Engineering, University of Pittsburgh, Pittsburgh, PA 15261, USA

Received 28 August 2006; Revised 13 February 2007; Accepted 25 June 2007

Recommended by Bülent Sankur

A new image sharing method, based on the reversible integer-to-integer (ITI) wavelet transform and Shamir's  $(r, m)$  threshold scheme is presented, that provides highly compact shadows for real-time progressive transmission. This method, working in the wavelet domain, processes the transform coefficients in each subband, divides each of the resulting combination coefficients into  $m$  shadows, and allows recovery of the complete secret image by using any  $r$  or more shadows ( $r \leq m$ ). We take advantages of properties of the wavelet transform multiresolution representation, such as coefficient magnitude decay and excellent energy compaction, to design combination procedures for the transform coefficients and processing sequences in wavelet subbands such that small shadows for real-time progressive transmission are obtained. Experimental results demonstrate that the proposed method yields small shadow images and has the capabilities of real-time progressive transmission and perfect reconstruction of secret images.

Copyright © 2007 C.-P. Huang and C.-C. Li. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. INTRODUCTION

With the rapid development of computer and communication networks, Internet has been established worldwide that brings numerous applications such as commercial services, telemedicine, and military document transmissions. Due to the nature of the network, Internet is an open system; to transmit secret application data securely is an issue of great concern. Security could be introduced in many different ways, for example, by image hiding and watermarking. However, the common weak point of them is that a secret image is protected in a single information carrier, and once the carrier is damaged or destroyed the secret is lost. If many duplicates are used to overcome this deficiency, the danger of security exposure will also increase [1, 2]. A secret image sharing method provides a viable solution. To process the received data efficiently is another problem. As transmission proceeds, the receiver may gradually access images with increased visual quality. If the received data is of no interest, the transmission can be terminated immediately to increase efficacy. Therefore, the functionality of progressive reconstruction is very essential to be built in the scheme. The goal is to develop an efficient secret image sharing method with progressive transmission capability.

Shamir [1] and Blakley [3] first proposed a concept of secret sharing called the  $(r, m)$  threshold scheme. In their scheme, a secret is shared by  $m$  shadows and any  $r$  shadows, where  $r \leq m$  can be used to reveal the secret while with less than  $r$  shadows the information about the secret cannot be obtained. Thien and Lin [2] developed a secret image sharing method based on Shamir's  $(r, m)$  threshold scheme. Their method permutes a secret image first to decorrelate pixels and then incorporates the  $(r, m)$  threshold scheme to process the image pixel wise or pattern wise in the spatial domain sequentially; hence, it may not be suitable for real-time progressive transmission. Each generated shadow is  $1/r$  the size of the original image for their lossy scheme and is over  $1/r$  for their lossless version [2]. Recently, Chen and Lin [4] developed a method of progressive image transmission for the secret image sharing [2]. Their method considers the division of an image into nonoverlapped sectors and applies a bit-plane scanning to rearrange the gray value information of each sector with several thresholds in controlling the reconstruction quality level to achieve the capability of progressive transmission. It tends to yield large shadow images due to its requirement of satisfactory functioning for every chosen threshold, thus reducing the efficiency of storage and

transmission. Since it works on a sector basis, the progression is localized to each sector; and it suffers from the blocking effects when images at low bit rate are recovered. Wang and Su [5] developed a secret image sharing method based on the Galois field. It has the advantage of producing small shadow images but does not have the progressive transmission capability. In comparison to these existing methods, the proposed method, working in the wavelet domain, has the advantage of both having small shadow images and progressive transmission capability at the same time. This is achieved by using the reversible integer-to-integer (ITI) wavelet transform and Shamir's  $(r, m)$  threshold scheme.

An integer-to-integer reversible wavelet transform maps an integer-valued image to integer-valued transform coefficients and provides the exact (*lossless*) reconstruction of the original image [6–9]. Its multiresolution representation is the same as usual, but can be fast computed with only integer addition and bit-shift operations. Most of the signal energy is concentrated in the low frequency bands and the transform coefficients therein are expected to be better magnitude-ordered as we move downward in the multiresolution pyramid in the same spatial orientation [6, 7, 10]. These properties are very important for the development of an image sharing method with real time progressive transmission. Instead of using permutation to decorrelate pixels prior to applying the  $(r, m)$  threshold scheme as in [2], we first apply ITI wavelet transform and then process transform coefficients in a preprocessing stage to decorrelate pixels (coefficients) and increase security. The preprocessing stage is performed on subband basis and the resulting coefficients in each subband are processed in a zigzag sequence from the smooth subband to detail subbands. The most important information of the smooth subband will be processed first and then the detail bands so that the progressive transmission can be obtained. In SPIHT [10], the progressive transmission is achieved by checking several times the transform coefficients. In the proposed method, the progressive transmission is enabled by ordering the importance of the subband information and checking the coefficients only one time to speed up the processing. The proposed method, based on the ITI wavelet transform, provides small shadows, lossless secret image reconstruction, and more importantly the capability of real time progressive transmission. In this method, a secret image will be transmitted by  $m$  distinct channels (shadows), any  $r$  shadows received in  $r$  channels (where  $r \leq m$ ) can be used to reveal the secret image while up to any  $r - 1$  channels intercepted by an adversary cannot reveal any secret. Also, it can tolerate up to  $m - r$  contaminated channels without affecting the lossless reconstruction of the secret image from the other  $r$  channels. A note should be made here that this method is significantly different from the multiple description coding (MDC) [11, 12]. Although both methods generate multiple subimages and utilize the information therein for image transmission over networks, our method addresses the issue of security protection of confidential images for transmission, while MDC does not consider the security question but emphasizes on multiple representations of an image for use in noisy channel transmission allowing

image reconstruction to continue even a packet is lost or severely contaminated.

The rest of the paper is organized as follows. The  $(r, m)$  threshold scheme is reviewed in Section 2. The proposed image sharing algorithm is described in Section 3. The experimental results are shown in Section 4. Security analysis is given in Section 5. Applications of the method are described in Section 6. Finally, the conclusions are summarized in Section 7.

## 2. PREVIOUS WORKS

According to Shamir's  $(r, m)$  threshold scheme [1], the secret  $D$  is divided into  $m$  shadows  $(D_1, D_2, \dots, D_m)$  and any  $r$  or more shadows can be used to reconstruct it. To split  $D$  into  $m$  pieces, a prime  $p$ , which is bigger than both  $D$  and  $m$ , is randomly selected and an  $(r - 1)$ th degree polynomial is chosen,

$$q(x) = (a_0 + a_1x + \dots + a_{r-1}x^{r-1}) \bmod p, \quad (1)$$

in (1),  $a_0 = D$ , and  $\{a_1, a_2, \dots, a_{r-1}\}$  are random numbers selected from numbers  $0 \sim (p - 1)$ . The pieces are obtained by evaluating

$$D_1 = q(1), \dots, D_i = q(i), \dots, D_m = q(m). \quad (2)$$

Note that  $D_i$  is a shadow. Given any  $r$  pairs from these  $m$  pairs  $\{(i, D_i); i = 1, 2, \dots, m\}$ , the coefficients  $a_0, a_1, a_2, \dots, a_{r-1}$  can be solved using Lagrange's interpolation, and hence the secret data  $D$  can be revealed. In Thien and Lin's work, they took  $a_0, a_1, a_2, \dots, a_{r-1}$  as the gray levels of  $r$  pixels in a secret image to generate  $m$  shadows.

An ITI reversible wavelet transform [6, 7] with a high computation speed and excellent energy compaction maps an integer-valued image to integer-valued smooth (scaling) coefficients and detail (wavelet) coefficients and provides the exact (*lossless*) reconstruction. It can be fast computed with only integer addition and bit-shift operations. The smooth coefficients have the same range of values as that of the input image and the detail coefficients have smaller absolute integer values than those of the input image.

## 3. THE PROPOSED IMAGE SHARING METHOD

In the proposed method described below, we take  $a_0, a_1, a_2, \dots, a_{r-1}$  as values of  $r$  processed transform coefficients to generate  $m$  shadows. A secret image is ITI wavelet transformed down to a selected scale level to form its multiresolution hierarchical representation. A preprocessing stage for wavelet transform coefficients in individual subbands is developed based on the strong intra-band correlation and small absolute values of the coefficients in the detail subbands. Thus, we expect to have small values of differences between neighboring coefficients in the smooth subband and small coefficients in the detail subbands. These are used in the preprocessing stage in the respective subbands to produce combination coefficients for use in the  $(r, m)$  threshold scheme. The sequence of the preprocessing stage starts from

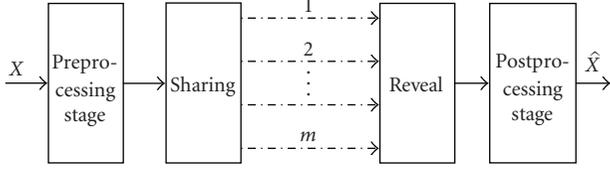


FIGURE 1: The block diagram of the proposed method.

the smooth subband and follows a zigzag path to the detail subbands in a hierarchical tree [10] such that the progressive transmission may be readily achieved. The block diagram of the proposed method is shown in Figure 1, where  $X$  denotes coefficients of the wavelet multiresolution representation of an image and  $\hat{X}$  the reconstructed wavelet transform coefficients.

### 3.1. The preprocessing stage

The wavelet transform coefficients in each subband are appropriately combined so as to decorrelate coefficients, prior to applying the  $(r, m)$  threshold scheme for enhancing security. Since the numbers (in images with 8-bit intensity levels) suitable for the  $(r, m)$  threshold scheme are from 0 to 255 [2], we need to take care of this requirement in the coefficient combination procedure. The combination process is designed by concatenating neighboring transform coefficients (or coefficients differences in the smooth subband) into one byte in case they are small enough or else scaling their values into the appropriate range. Then the size of the resulting combination coefficients is reduced and its range is adjusted.

Consider the smooth subband with scaling coefficients  $S = \{s_{u,v}\}$  and coefficient differences  $DS = \{ds_{u,v}\}$ . At location  $(u, v)$ , the coefficient difference is defined by

$$ds_{u,v} = \begin{cases} s_{u,v}, & \text{if } u = 0, v = 0, \\ s_{u,v} - s_{(u-1),v}, & \text{if } u \neq 0, v = 0, \\ s_{u,v} - s_{u,(v-1)}, & \text{otherwise.} \end{cases} \quad (3)$$

A sequence of combination numbers  $C_{\text{com}} = \{c_{\text{com}}\}$  are generated, referring to differences  $DS$ , in the following steps.

- (1) Divide the array of differences  $DS$  into nonoverlapping blocks, each block contains  $2 \times 2$  neighboring differences.
- (2) Process each block from left to right and top to bottom.
- (3) In each block, the coefficient differences are combined as follows: (i) if the values of four differences are all not less than  $-2$  and not greater than  $1$ , then these four differences are processed together by adding  $2$  to each difference and concatenating them into a new byte  $c_{\text{com}}$ . Note that the concatenation is done by *bitshift* and *bitor* operators. (ii) If the values of the successive two differences (in either upper row or lower row of a block) are both not less than  $-4$  and not greater than

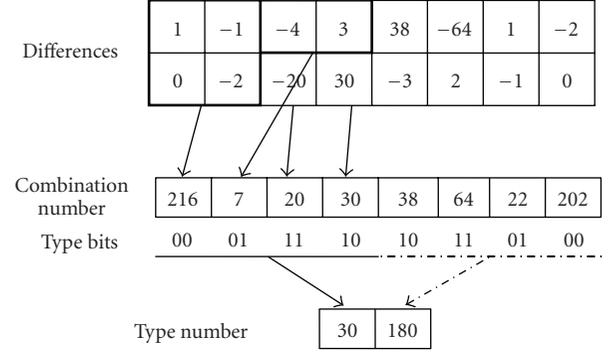


FIGURE 2: An illustration of the preprocessing stage.

3, then these two differences are processed together by adding  $4$  to each difference and concatenating them into a new byte  $c_{\text{com}}$ . (iii) If the values of four differences do not satisfy the condition (i) or (ii), then each coefficient difference is processed separately to form a new byte  $c_{\text{com}}$  by multiplying itself with its sign.

- (4) The new byte  $c_{\text{com}}$  generated in step (3) is assigned sequentially in a sequence of combination numbers  $C_{\text{com}} = \{c_{\text{com}}\}$ . Note that the value of  $c_{\text{com}}$  is between  $0$  and  $255$ .
- (5) Use two bits to record the type of a new byte in step (3) as follows:  $00$  and  $01$  for concatenation of four and two differences, respectively;  $10$  and  $11$  for a positive and a negative valued byte, respectively. Every four consecutive such type bits are concatenated to form a byte called  $t_{\text{com}}$ . Note that the value of  $t_{\text{com}}$  is between  $0$  and  $255$ .
- (6) The byte  $t_{\text{com}}$  generated in step (5) is recorded sequentially in a sequence of type numbers  $T_{\text{com}} = \{t_{\text{com}}\}$ .

For illustration of the wavelet transform coefficient preprocessing stage, let us consider an array of transform coefficients of size  $2 \times 8$  (or coefficient differences in the case of a smooth subband) as shown in Figure 2. The first block meets the condition (i) so that the four differences  $\{1, -1, 0, -2\}$  in the block are each added by  $2$  to give  $\{3, 1, 2, 0\}$ . These four numbers  $\{3, 1, 2, 0\}$  are processed together by concatenation using *bitshift* and *bitor* operators as follows. The four data in their binary representation are *bitshift* first to give  $\{11000000, 00010000, 00001000, 00000000\}$  and followed by *bitor* to get  $c_{\text{com}} = (11011000)_2 = 216$ . Two bits  $00$  are given as the type value to record this block. The next block meets the condition (ii) for the upper row and condition (iii) for the lower row. The two differences in the upper row satisfies condition (ii) so that each of the two differences in the block  $\{-4, 3\}$  is added by  $4$  to give  $\{0, 7\}$ . Then  $\{0, 7\}$  is processed by concatenation using *bitshift* and *bitor* operators. The two data are *bitshift* first to get  $\{00000000, 00000111\}$  with binary representation and followed by *bitor* to get  $c_{\text{com}} = (00000111)_2 = 7$ . Two bits  $01$  are given as the type value to record the upper row of the block. The two differences  $\{-20, 30\}$  in the lower row satisfies condition (iii) so that

they are processed separately to get 20 and 30. The two bits 11 and 10 are given as type values to record these two differences respectively in the lower row of the block. The other blocks are processed in the same way. The type number  $t_{\text{com}}$  is obtained by concatenating every four consecutive 2-bit type bits as indicated in Figure 2.

The similar combination process is used for coefficients in detail subbands, referring to wavelet coefficients  $S$ . The inverse combination can be easily done by following the reverse steps in the postprocessing stage.

### 3.2. The sharing phase

The sequence of type numbers  $T_{\text{com}}$  and the sequence of combination numbers  $C_{\text{com}}$  are each divided into nonoverlapping sharing blocks each containing a sequence of  $r$  number. For each sharing block  $b$ , a  $(r - 1)$ th degree polynomial is used as in [2] except here the prime number  $p = 257$ ,

$$q_b(x) = (a_0 + a_1x + \dots + a_{r-1}x^{r-1}) \bmod 257, \quad (4)$$

where  $a_0, a_1, a_2, \dots, a_{r-1}$  are  $r$  numbers of the sharing block. Evaluate

$$D_1 = q_b(1), \dots, D_i = q_b(i), \dots, D_m = q_b(m). \quad (5)$$

The  $m$  output numbers  $q_b(1), \dots, q_b(i), \dots, q_b(m)$  of this sharing block  $b$  are placed sequentially in the  $m$  shadow coefficients. In this case, the possible values of the output are  $0 \leq q_b(i) \leq 256, i = 1, \dots, m$ . The problem is that the value of a byte coefficient is in the range from 0 to 255 while in output numbers there may be 256. If the output values are 255 and 256, this problem can be dealt with by storing 255 with an extra bit of 0 or 1 (for output value of 255 or 256, resp.) stored in the following byte. In order to provide for progressive transmission and to establish a traceable set of coefficient combination numbers  $C_{\text{com}}$ , the type numbers  $T_{\text{com}}$  and the byte for the extra bit are stored as an overhead. Note that  $r$  type combination numbers  $t_{\text{com}}$  are associated with the corresponding  $4r$  coefficient combination numbers,  $c_{\text{com}}$ . The prime number  $p$  is selected to be 257, using the same rationale as that in [1, 2], which is the smallest prime number greater than the largest number 255 possible after the preprocessing stage. For a relatively large value of  $p$  considered here, a practical choice of  $r$  and  $m$  will be  $r < m \ll p$ . For security of sharing, we would like to have  $r$  to be more than just a couple, but be limited in connection with limiting  $m$  to reduce the computation involved and to avoid the use of too many channels. The  $r$  and  $m$  are chosen based on the application on hand. For example, in the  $(r = 4, m = 6)$  threshold scheme, let us consider a system consisting of one dealer and six participants, the dealer distributes a secret image into  $m = 6$  shares and each participant holds one share. Later, if  $r = 4$  shares are received, the secret image can be revealed. If less than 4 shares are received, then no information about the secret image can be revealed.

The sharing process is described below:

- (1) from the preprocessing stage, we get combination numbers  $C_{\text{com}}$  and type numbers  $T_{\text{com}}$ ;

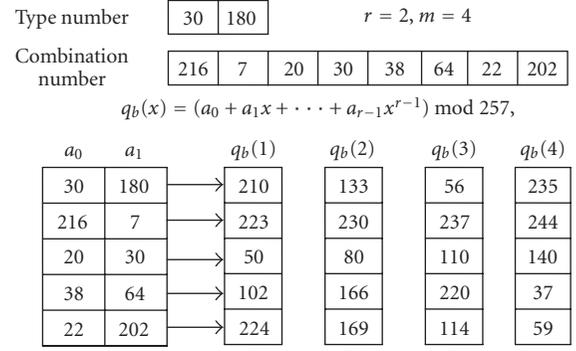
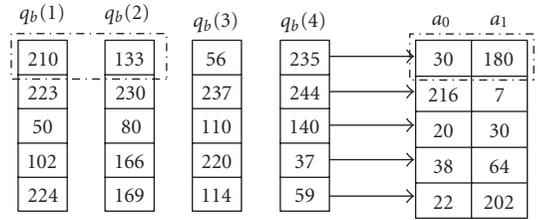


FIGURE 3: An illustration of the sharing phase.



Any  $r = 2$  out of  $m = 4$  shadows can reveal  $a_0, a_1$

$$\left. \begin{array}{l} 210 = (a_0 + a_1) \bmod 257 \\ 133 = (a_0 + 2a_1) \bmod 257 \end{array} \right\} \rightarrow a_0 = 30, a_1 = 180$$

FIGURE 4: An illustration of the reveal phase.

- (2) pick  $r$  consecutive numbers from  $T_{\text{com}}$  and  $4r$  consecutive numbers from  $C_{\text{com}}$  to form five sharing blocks each containing  $r$  numbers;
- (3) apply the sharing equations (4) and (5) to the picked sharing block to generate  $m$  output shares for the  $m$  shadows. If the output values are less than 255, store the generated output shares in the shadows. If an output value is 255 or 256, then store the coefficient 255 in the shadow coefficients and an extra bit 0 for 255 and 1 for 256 is stored in a list that follows;
- (4) go to step (2) until all combination numbers are processed.

An illustration of the sharing phase is shown in Figure 3 using the type numbers and the combination numbers obtained from the illustration in Figure 2. Without loss of generality, consider  $r = 2$  and  $m = 4$ , that is, consider two numbers as polynomial coefficients in the sharing equation (4) and four output numbers  $q_b(1), q_b(2), q_b(4), q_b(5)$  as output shares for four shadows. Take  $a_0 = 30$  and  $a_1 = 180$ , the shares are  $q_b(1) = (30 + 180) \bmod 257 = 210$ ,  $q_b(2) = 133$ ,  $q_b(3) = 56$ , and  $q_b(4) = 235$ . The other shares are evaluated in the same way using the other coefficients as shown in Figure 3.

### 3.3. The reveal phase

The coefficient combination numbers can be revealed by any  $r$  out of  $m$  shadows via the following steps.

- (1) Take one pixel (element) from each of the  $r$  shadows to form a shadow block sequentially from left to right and top to bottom.
- (2) Use these  $r$  shares and apply Lagrange's interpolation to solve for the values of  $a_0, a_1, a_2, \dots, a_{r-1}$  in (4).
- (3) Steps (1) and (2) are processed for every 5 shadow blocks with one type combination block and 4 coefficient combination blocks. In case any value of  $q_b(i)$  is 255 in these 5 blocks, the following 6th shadow block is examined for the corresponding extra bit (0 or 1) to be added back.
- (4) Repeat steps (1) to (3) until all pixels of the  $r$  shadows are processed. The whole set of coefficient combination numbers is reconstructed.

An illustration of the reveal phase is shown in Figure 4 using the shares obtained from the illustration given in Figure 3 for  $r = 2$  and  $m = 4$ . The combination number can be revealed by any 2 out of 4 shadows. For example, take two shares  $q_b(1), q_b(2)$  and apply Lagrange's interpolation to solve for two values  $a_0$  and  $a_1$  from (6):

$$(a_0 + a_1) \bmod 257 = 210, \quad (a_0 + 2a_1) \bmod 257 = 133. \quad (6)$$

It gives  $a_0 = 30$  and  $a_1 = 180$  as expected. The other coefficient combination numbers can be revealed in the same way as shown in Figure 4.

## 4. EXPERIMENTAL RESULTS

Four images (Lena, Jet, Monkey, and Peppers), each has  $512 \times 512$  pixels with 8 bits per pixel, were used in the experiment. The ITI wavelet derived from Daubechies' 5/3 biorthogonal wavelet, 6-level decomposition, and the  $(r, m)$  threshold scheme with  $r = 4$  and  $m = 6$  were used. The small shadow sizes produced by the proposed method are shown in Figure 5(a) in comparison to those obtained by Thien and Lin's (TL's) method [2], Chen and Lin's (CL's) method [4] and Wang and Su's (WS's) method [5], respectively. The proposed method has smaller shadow images when comparing with TL's and CL's methods in all cases. Our method without coding (WO) has larger shadow images than those of WS's method that has been coded prior to inputting to the sharing phase. In order to have a fair comparison, the proposed method was also encoded either with Huffman coding (WHu) or with arithmetic coding (WAr) [13] before the data input to the sharing phase as the WS's method did. The results indicate that our method encoded with Huffman coding (WHu) has slightly smaller shadow images than those of WS's method, and the proposed method encoded with arithmetic coding (WAr) has significantly smaller shadow images than those of WS's method. The progressive transmission and reconstruction performances are compared to those obtained by Chen and Lin's (CL's) method [4]. The three cases

of CL's method described in [4] are as follows: case (1), with three thresholds ( $k = 3$ ) and settings  $r_1 = 3, r_2 = 4$ , and  $r_3 = 5$  for  $m = 6$ , case (2), with five thresholds ( $k = 5$ ) and settings  $r_1 = 3, r_2 = 4, r_3 = 5, r_4 = 5$ , and  $r_5 = 5$  for  $m = 6$ , and case (3), with five thresholds ( $k = 5$ ) and settings  $r_1 = 3, r_2 = 3, r_3 = 3, r_4 = 4$ , and  $r_5 = 5$  for  $m = 6$ . As shown in Figure 5(b), the experimental results of the proposed method are compared favorably to those of CL's method. The proposed method needs less bytes of shadow images than the original image data to achieve lossless reconstruction of the original image, while CL's method requires more bytes of shadow images than the original image data ( $512 \times 512$  bytes). In Figures 5(c) and 5(d), the experimental results on reconstructed image quality (PSNR) of four test images at different bit rates are shown, the PSNR of the reconstructed images by the proposed method with arithmetic coding is compared with those obtained by CL's method for all three cases. Our method gave higher quality (PSNR) reconstructed images. The performance of the proposed method on Peppers image is shown in Figure 6 for visual illustration. Figure 6(a) is the original Peppers image and Figure 6(b) shows the lossless reconstruction using four of the six shadows shown in Figure 6(e). The result of the preprocessing stage is shown in Figure 6(c). The histograms of the original image and of the result of the preprocessed data are shown in Figure 6(d) left part and right part, respectively. The latter appears more evenly distributed across a broad range in the middle, and the visual observations indicate that the data after the preprocessing stage are significantly decorrelated. At the bit rate of 2.0 bpp, our reconstructed image is shown in Figure 7(a) in comparison to the reconstruction obtained by applying CL's method as shown in Figure 7(b). As expected, the proposed method has better visual quality of the reconstructed image at the lower bit rate. In another experiment on map images, as will be discussed in Section 6, the progressive reconstruction of the proposed method is shown in Figures 12 and 13.

In order to have an idea about the transmission performance of the proposed method when channel interference (noise or mis-synchronization) occurs, we illustrate the performance of the method using  $r = 4$  and  $m = 6$ . If the noisy or misalignment channels are no more than  $(m - r)$  channels while  $r$  channels are received free from noise, the image can be perfectly reconstructed without being affected by the interference. For interference occurred in the  $r$  channels, let us consider an ordinary communication system for binary pulse amplitude modulation (PAM) baseband signals with a controllable additive white Gaussian noise [14] or misalignment steps (bits). The transmission characteristic of this communication system [14] with bit-error rate (BER) versus signal-to-noise ratio (SNR,  $E_b/N_0$ , dB) is shown in Figure 8(a), where  $E_b$  is energy per bit and  $N_0$  is noise spectral density. Such a controlled additive white Gaussian noise was added in every channel and the shadow images were transmitted over the channels bit by bit. The number of error bits was measured at every controlled noise level to obtain bit-error rates for four test images during their shadow transmission. We used the received shadow data to reconstruct

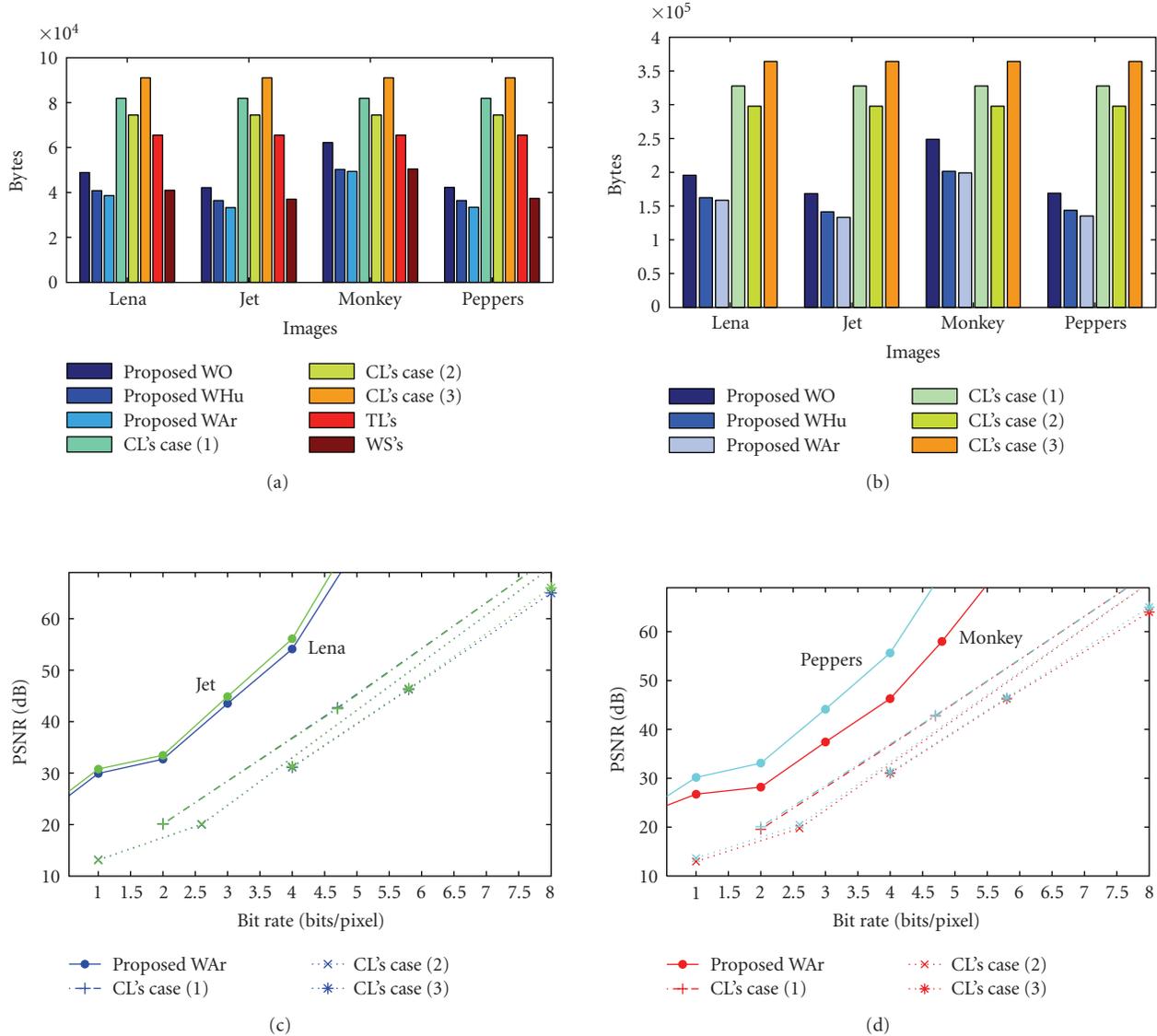


FIGURE 5: Performance of shadow size and reconstruction quality of the proposed method on four test images (Lena, Jet, Monkey, and Peppers): (a) shadow size comparison (Bytes), (b) number of bytes used for lossless reconstruction, (c) quality (PSNR, dB) of reconstructed images (Lena, Jet) at different bit rate, and (d) quality (PSNR, dB) of reconstructed images (Monkey, Peppers) at different bit rate.

the four images and computed peak signal-to-noise ratios (PSNR, dB) corresponding to each bit-error rate for these four images, the results are shown by curves in Figure 8(b). For visual evaluation, the reconstructed Peppers image of PSNR 16.04 dB at the bit-error rate of  $8 \times 10^{-2}$ , the reconstructed image of PSNR 25.10 dB at the bit error rate of  $2.4 \times 10^{-3}$ , and the reconstructed image of PSNR 35.10 dB at the bit error rate of  $2 \times 10^{-4}$  are shown in Figures 8(c), 8(d), and 8(e), respectively. The mis-synchronization problem was evaluated by the BER and misalignment steps (bits). The average BER versus misalignment steps (bits) of the four test images is shown in Figure 8(f). The average over this range is 0.4283. For visual evaluation, the reconstructed Pep-

pers image with PSNR of 5.67 dB at 1-bit misalignment from the starting point is shown in Figure 8(g). It indicates that the method is very sensitive to mis-synchronization from the beginning. Since the proposed method has the progressive transmission capability, it should provide some reasonable visual quality if the misalignment occurs in the middle of the transmission. Three reconstructed Peppers images with PSNR of 11.88 dB, 24.16 dB, and 30.15 dB are shown in Figures 8(h), 8(i), and 8(j), when 1-bit misalignment occurred after 5 percent of the shadow data was transmitted, when 8-bis misalignment occurred after 20 percent of the data was transmitted, and when 10-bits misalignment occurred after 50 percent of the data was transmitted, respectively.

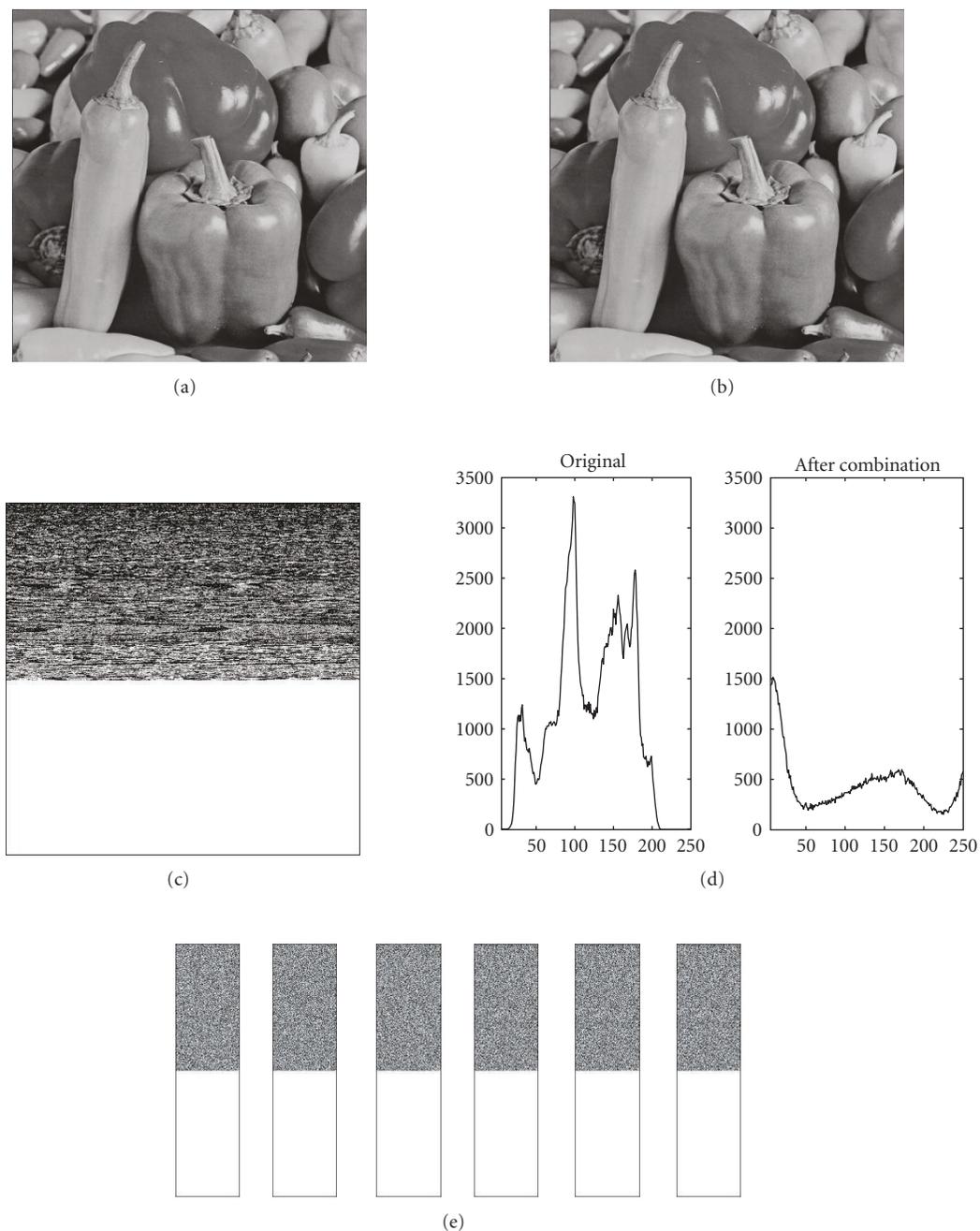


FIGURE 6: Illustration of the results of various processing phases of the Peppers image: (a) the original Peppers image, (b) the reconstructed image using four out of six shadows in (e), (c) the result of the preprocessing stage, (d) histogram of the original image and histogram of the combination coefficient image resulted from the preprocessing, and (e) shadows generated by the proposed method with  $r = 4$  and  $m = 6$ .

These results indicate that the shadow data from the proposed method can be transmitted over the channel of low-to-moderate noise level (e.g., bit-error rate smaller than  $10^{-3}$ ). It also indicates that the method may perform well if any mis-synchronization occurs after the first portion of the data has been transmitted. Its performance under interference will be enhanced when the channel coding is used in the transmission system as discussed in [15–17].

## 5. SECURITY ANALYSIS

A security analysis of the proposed method has been performed similar to what was done in [2] to ascertain that the method has the security property that “any  $r - 1$  or less shadows cannot provide sufficient information to reveal the secret image.” Note that our method utilizes ITI wavelet transform representation of the image and combines the wavelet coefficients prior to the sharing process. Without loss of generality,



FIGURE 7: Reconstructed image at the bit rate of 2.0 bpp, (a) using our method with PSNR of 32.61 dB and (b) using CL's method with PSNR of 20.08 dB.

let us inspect how coefficient combination numbers and type combination numbers (or coefficients  $a_0, \dots, a_{r-1}$ ) can be revealed. From (4), to reveal the  $r$  coefficients of the polynomial  $q_b(x)$ , we need  $r$  equations. If we only have  $(r-1)$  shadow images from which we get  $q_b(1), q_b(2), \dots, q_b(r-1)$ , we can only set up  $(r-1)$  equations

$$\begin{aligned} q_b(1) &= (a_0 + a_1 + \dots + a_{r-1}) \bmod 257, \\ q_b(2) &= (a_0 + 2a_1 + \dots + 2^{r-1}a_{r-1}) \bmod 257, \\ &\vdots \\ q_b(r-1) &= (a_0 + (r-1)a_1 + \dots + (r-1)^{r-1}a_{r-1}) \bmod 257, \end{aligned} \quad (7)$$

there are 257 possible solutions in solving for  $r$  unknown coefficients using only the above  $r-1$  equations, and hence the probability of guessing the correct solution is  $1/257$  if the shadow images have uniformly distributed intensity levels. There are  $t$  polynomials for an image with  $t$  sharing blocks, and hence the probability of obtaining the correct image is  $(1/257)^t$ . For example, for a  $512 \times 512$  secret image, if  $r = 2$ , there are about 100 000 polynomials to be involved. The probability of guessing the right pixel values of shadow images in the proposed scheme is  $(1/257)^{100,000}$  which is extremely small. An intruder has only this near zero probability to get the correct coefficient combination numbers, not to mention the difficulty to reconstruct the original image. The reconstructed image of the example on Peppers (with  $r = 2, m = 4$ ) is shown in Figure 9, using one valid shadow image and one randomly estimated shadow image. This result indicates that there is practically no correlation between the secret image (the original Peppers) and the reconstructed image using less than  $r$  valid shadow images.

Since the above security analysis of the sharing method is based on the assumption of uniformly distributed intensity levels of shadow images, it needs an experimental justification. Let us consider the normalized histogram of a shadow image with intensity levels  $\{x_i, i = 0, 1, \dots, n\}$  versus the

numbers of occurrences of  $x_i$  normalized by the total number of occurrences,  $\{f(x_i)$  versus  $x_i, i = 0, 1, \dots, n\}$ .  $f(x_i)$  is thus the probability of occurrences of  $x_i$ . Let  $\bar{f}$  be the mean value of the normalized histogram

$$\bar{f} = \frac{1}{n+1} \sum_{i=0}^n f(x_i) \quad (8)$$

and let  $\sigma$  be the estimated standard deviation

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=0}^n (f(x_i) - \bar{f})^2}. \quad (9)$$

For a uniform distribution,  $f(x_i)$  should be equal to  $\bar{f}$  for all  $x_i$ . The degree of distribution uniformity may be measured in terms of the ratio of standard deviation to mean ( $\sigma/\bar{f}$ ). The smaller the  $\sigma/\bar{f}$ , the closer the histogram is to a uniform distribution. The same four test images were used in the experimental evaluation. The average value of the ratio of standard deviation to mean for  $m$  shadow image histograms of each test image using the proposed method is shown in Figure 10 in comparison to those obtained by Thien and Lin's (TL's) method [2] and Chen and Lin's (CL's) method [4]. The proposed method has significantly smaller average values of  $\sigma/\bar{f}$  in the experimental study. This supports the hypothesis that histograms of the shadow images are almost uniformly distributed and the probability of guessing the right combination coefficients in the proposed scheme will be extremely small, so that our method is very secure. For visual comparison, histograms of the shadow images of Jet image obtained by using the proposed method, TL's method and CL's method are shown in Figures 11(a), 11(b), and 11(c), respectively. In Figures 11(a) and 11(b), the parameters used were  $r = 4$  and  $m = 6$ , and in Figure 11(c), the case (1) was investigated. Note that for a fair comparison the permutation process was not applied to any method in this experiment. This verifies the adequacy of the security analysis discussed above.

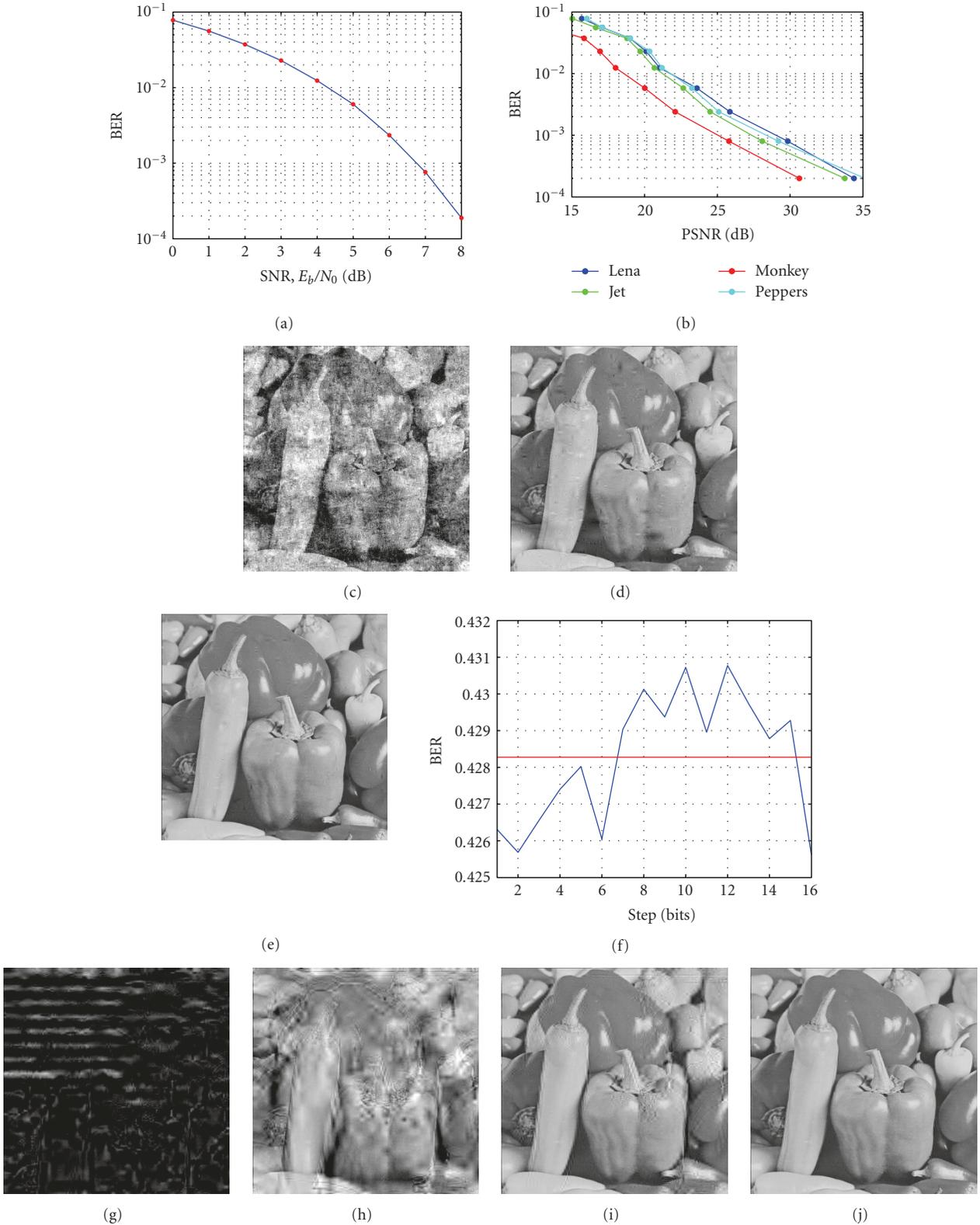


FIGURE 8: Performance of the proposed method under interference with channel noise or mis-synchronization: (a) performance of an ordinary communication system, (b) quality of the reconstructed images (PSNR, dB) at different bit-error rate, (c) reconstructed Peppers image with PSNR of 16.04 dB at bit-error rate of  $8 \times 10^{-2}$ , (d) reconstructed Peppers image with PSNR of 25.10 dB at bit-error rate of  $2.4 \times 10^{-3}$ , (e) reconstructed Peppers image with PSNR of 35.14 dB at bit-error rate of  $2.0 \times 10^{-4}$ , (f) average bit-error rate at different misalignment steps, (g) reconstructed Peppers image with PSNR of 5.67 dB for 1 bit misalignment from the beginning, (h) reconstructed Peppers image with PSNR of 11.88 dB for 1-bit misalignment after 5 percent of the shadow data was transmitted, (i) reconstructed Peppers image with PSNR of 24.16 dB for 8-bit misalignment after 20 percent of the data was transmitted, and (j) reconstructed Peppers image with PSNR of 30.15 dB for 10-bit misalignment after 50 percent of the data was transmitted.

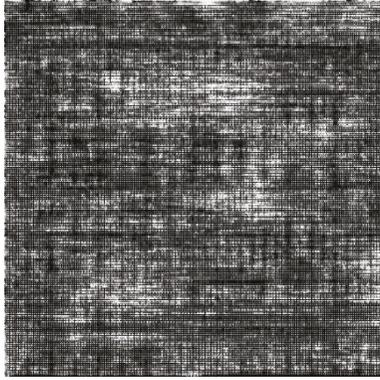


FIGURE 9: The reconstructed Peppers image by using  $r - 1$  valid shadow images in the case of  $(r = 2, m = 4)$ .

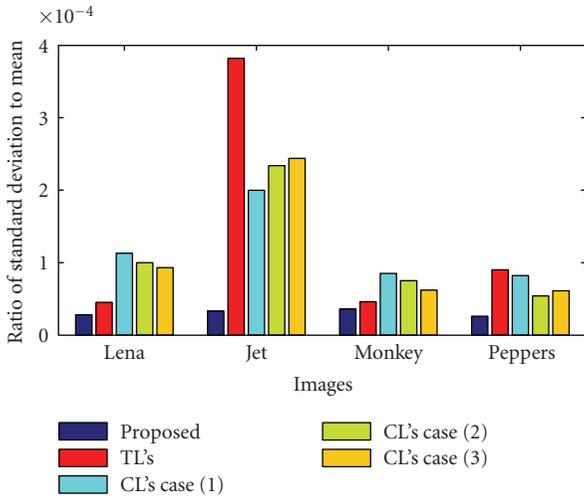


FIGURE 10: Average value of the ratio of standard deviation to mean of histograms of six shadow images for test images Lena, Jet, Monkey and Peppers.

## 6. APPLICATIONS

We consider to apply the proposed method to secret image telebroadcasting (e.g., military maps) to illustrate one of the practical applications of the proposed method. Firstly, apply integer wavelet transform and Shamir's  $(r, m)$  threshold scheme to divide each military image into several shadows and distribute them to several different sites. It assures that the secret images are protected securely. Since the quantities of military maps used in a war are tremendous and the proposed method produces small shadows, it has the advantage of saving storage space. Secondly, apply the reveal procedure to progressively reconstruct the related military maps. Since the proposed method has progressive transmission capability, during the reconstruction soldiers (viewers) may quickly skip irrelevant maps and can find the desired maps efficiently. Two military images from [18] are used to demonstrate this application of the proposed method. If the desired map is not Map1 in Figure 12, a soldier may skip the image at the glance

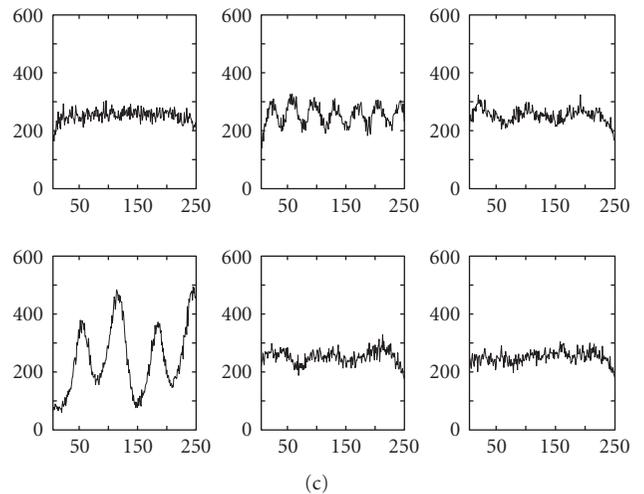
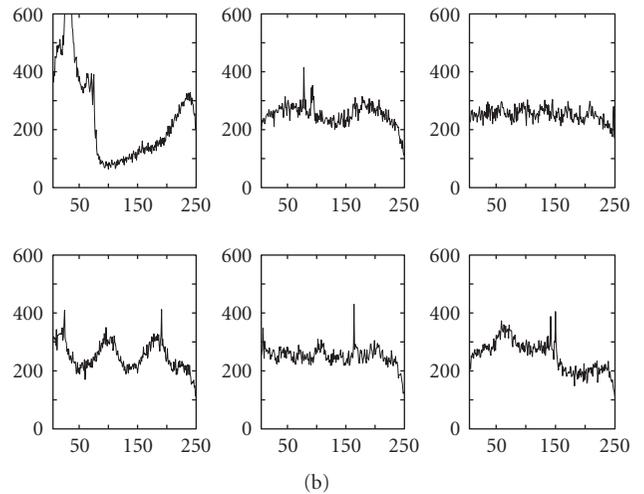
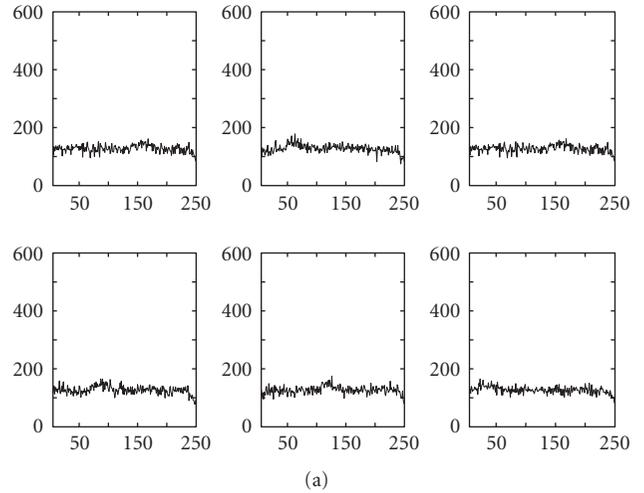


FIGURE 11: Shadow image histograms of the Jet image: (a) using the proposed method, (b) using TL's method, and (c) using CL's method of case (1).

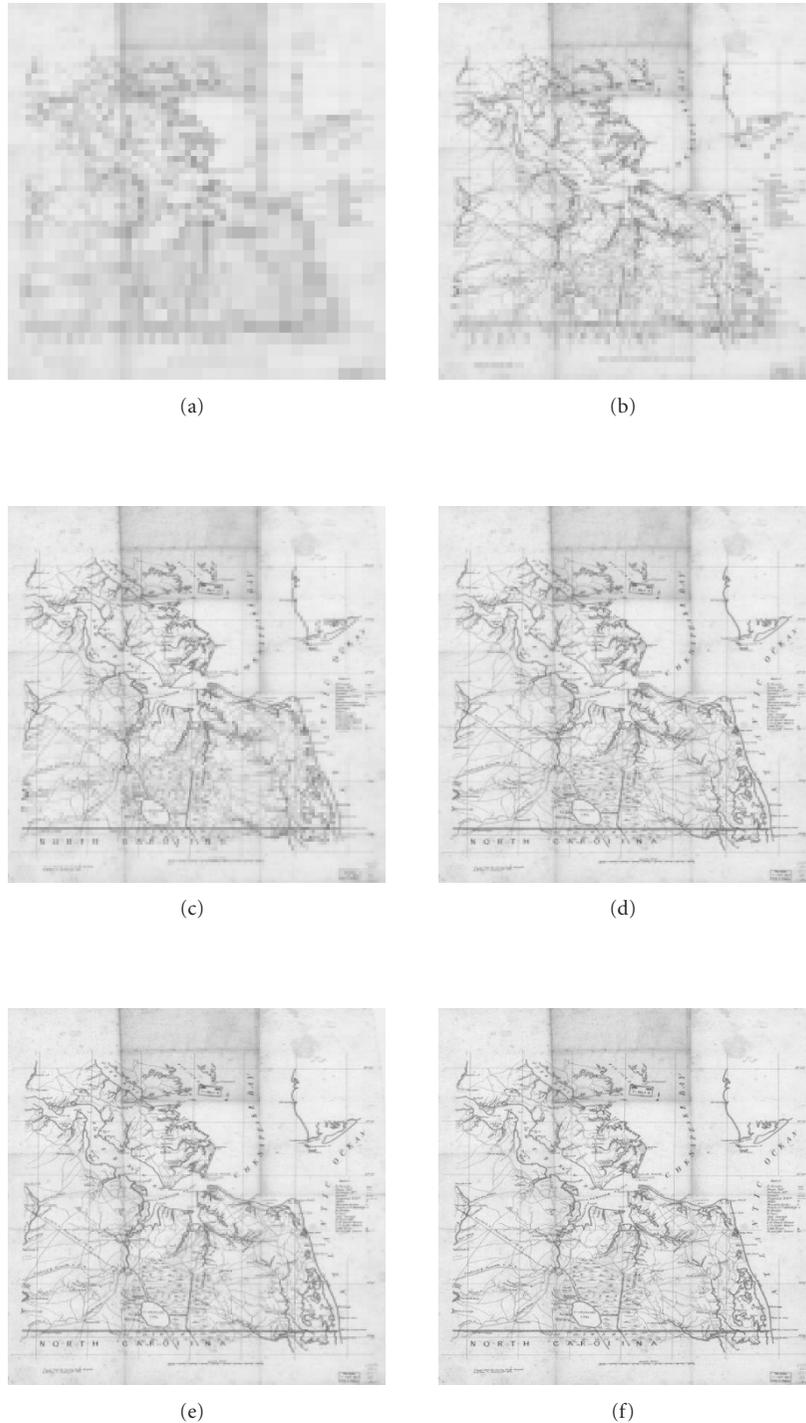


FIGURE 12: Progressive reconstruction of Military Map1 using any 4 out of 6 shadows and with the following percentages of coefficients, and the resulting PSNR: (a) 1%, 21.38 dB, (b) 5%, 23.72 dB, (c) 15%, 25.73 dB, (d) 25%, 28.27 dB, (e) 50%, 32.15 dB, (f) 100%, lossless.

of the reconstructed image of the lowest possible resolution, that is, in Figure 12(a). The soldier will look for the target image Map2 in Figure 13, and will keep progressive reconstruction to the required quality even to the perfect reconstruction should the received shadow images be not corrupted by any channel noise.

## 7. CONCLUSIONS

In this paper, a new method based on the reversible ITI wavelet transform to share a secret image has been presented. By taking advantages of transform coefficient magnitude decay and excellent energy compaction in wavelet

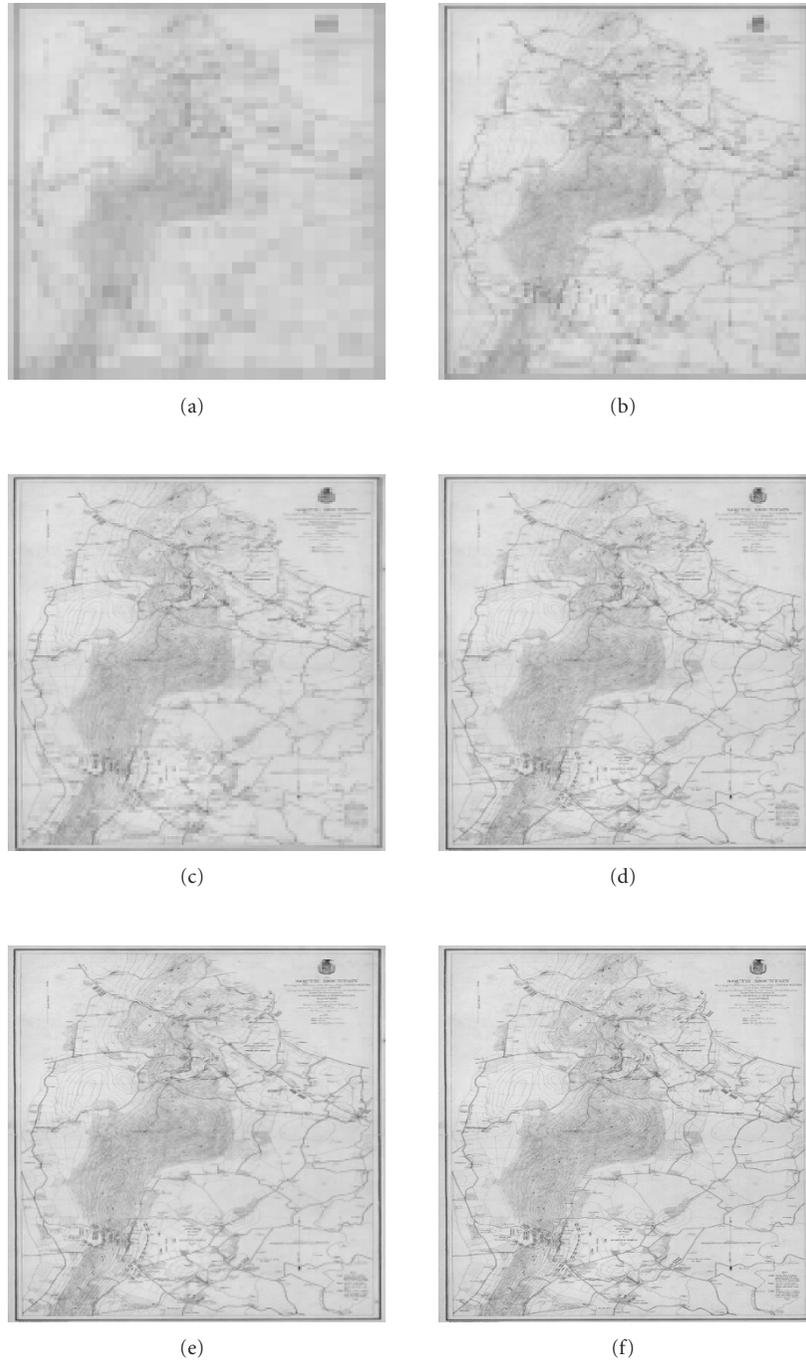


FIGURE 13: Progressive reconstruction of Military Map2 using any 4 out of 6 shadows and with the following percentages of coefficients, and the resulting PSNR: (a) 1%, 22.19 dB, (b) 5%, 23.48 dB, (c) 15%, 24.85 dB, (d) 25%, 28.09 dB, (e) 50%, 31.21 dB, (f) 100%, lossless.

multiresolution representation, coefficient combination procedures and processing sequences are developed for use in applying the  $(r, m)$  threshold scheme to generate shadows for image sharing. It results in small shadow images, perfect reconstruction, and the capability for progressive transmission. The effectiveness of the proposed method is demonstrated by experimental results on test images. In comparison to the methods in [2, 4, 5], the proposed method has advantages of

providing both progressive transmission and small shadow images simultaneously. The security analysis result indicates that the method has the desired security property that “any  $r - 1$  or less shadows cannot provide sufficient information to reveal the secret image.” When considering the security quality in terms of distribution uniformity of histograms of shadow images, the proposed method is more secure (nearly uniform) than the existing methods in [2, 4].

## ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions on the earlier version of the manuscript. This research is supported in part by the National Science Council, ROC under the grant NSC 95-2221-E-130-019.

## REFERENCES

- [1] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.
- [2] C.-C. Thien and J.-C. Lin, "Secret image sharing," *Computers and Graphics*, vol. 26, no. 5, pp. 765–770, 2002.
- [3] G. R. Blakley, "Safeguarding cryptographic keys," in *Proceedings of AFIPS National Computer Conference*, vol. 48, pp. 313–317, New York, NY, USA, June 1979.
- [4] S.-K. Chen and J.-C. Lin, "Fault-tolerant and progressive transmission of images," *Pattern Recognition*, vol. 38, no. 12, pp. 2466–2471, 2005.
- [5] R.-Z. Wang and C.-H. Su, "Secret image sharing with smaller shadow images," *Pattern Recognition Letters*, vol. 27, no. 6, pp. 551–555, 2006.
- [6] H. Kim and C. C. Li, "Lossless and lossy image compression using biorthogonal wavelet transforms with multiplierless operations," *IEEE Transactions on Circuits and Systems II*, vol. 45, no. 8, pp. 1113–1118, 1998.
- [7] A. Zandi, J. Allen, E. Schwartz, and M. Boliek, "CREW: compression with reversible embedded wavelets," in *Proceedings of the 5th Data Compression Conference*, pp. 212–221, Snowbird, Utah, USA, March 1995.
- [8] A. R. Calderbank, I. Daubechies, W. Sweldens, and B.-L. Yeo, "Wavelet transforms that map integers to integers," *Applied and Computational Harmonic Analysis*, vol. 5, no. 3, pp. 332–369, 1998.
- [9] M. D. Adams and R. K. Ward, "Symmetric-extension-compatible reversible integer-to-integer wavelet transforms," *IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2624–2636, 2003.
- [10] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 243–250, 1996.
- [11] V. A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 821–834, 1993.
- [12] Y. Wang, A. R. Reibman, and S. Lin, "Multiple description coding for video delivery," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 57–70, 2005.
- [13] K. Sayood, *Introduction to Data Compression*, Morgan Kaufmann, San Francisco, Calif, USA, 2nd edition, 2000.
- [14] J. G. Proakis and M. Salehi, *Communication Systems Engineering*, Prentice-Hall, Englewood Cliffs, NJ, USA, 2nd edition, 2001.
- [15] P. G. Sherwood and K. Zeger, "Progressive image coding for noisy channels," *IEEE Signal Processing Letters*, vol. 4, no. 7, pp. 189–191, 1997.
- [16] V. Chande and N. Farvardin, "Progressive transmission of images over memoryless noisy channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 850–860, 2000.
- [17] N. V. Boulgouris, N. Thomos, and M. G. Strintzis, "Transmission of images over noisy channels using error-resilient

wavelet coding and forward error correction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 12, pp. 1170–1181, 2003.

- [18] The Library of Congress website, <http://memory.loc.gov/ammem/gmdhtml/milhome.html>.

**Chin-Pan Huang** was born in 1959 in Taiwan. He received the B.S. and M.S. degrees in electrical engineering from Chung Cheng Institute of Technology, Taiwan, in 1981 and 1985, respectively. In 1996, he received the Ph.D. degree in electrical engineering from University of Pittsburgh, Pa, USA. From 1996 to 2002, he was an Associate Scientist of the Electronic System Division in Chung Shan Institute of Science and Technology. He then joined the Department of Computer and Communication Engineering at Ming Chuan University in August 2002 and is currently an Assistant Professor there. His recent research interests include digital signal/image processing, data compression, and pattern recognition. He is a Member of IEEE.



**Ching-Chung Li** received his B.S. degree from the National Taiwan University, Taipei, in 1954, and his M.S. and Ph.D. degrees from Northwestern University, Evanston, ILL, in 1956 and 1961, respectively. He is presently a Professor of Electrical Engineering and Computer Science at the University of Pittsburgh, Pittsburgh, Pa. He was a Visiting Associate Professor of Electrical Engineering at the University of California, Berkeley, in the Spring of 1964, and a Visiting Principal Scientist at the Biodynamics Laboratory, Alza Corporation, Palo, Calif in the summer of 1970. On his sabbatical leaves, he was with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology in the Fall of 1988, and with Carnegie Mellon University at the Robotics Institute in the spring of 1998 and at the Advanced Multimedia Processing Laboratory in the spring of 2006. His research interests are in pattern recognition, image processing, biocybernetics, and applications of wavelet transforms. He is a Fellow of IEEE and a Fellow of IAPR.

