

## Research Article

# An Efficient Prediction-and-Shifting Embedding Technique for High Quality Reversible Data Hiding

Wien Hong

*Department of Information Management, Yu Da University, Miaoli, 361, Taiwan*

Correspondence should be addressed to Wien Hong, wienhong@ydu.edu.tw

Received 27 November 2009; Revised 9 March 2010; Accepted 31 March 2010

Academic Editor: Jin-Hua She

Copyright © 2010 Wien Hong. 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.

The embedding capacity of a histogram-based reversible data hiding technique is primarily determined by the peak height of the histogram. Recently, some studies have tried to embed data in the histogram of prediction errors by modifying the error values and have better embedding efficiency. However, these methods offer no selective embedment mechanism to exclude the positions where the modification in the embedding operation contributes no capacity but merely degrade the image quality. In this paper, a novel coding method for reversible data hiding is presented. A two-stage prediction algorithm that fully exploits the pixel correlations is employed to create more embeddable spaces, and a selective embedment mechanism is used to enhance the image quality. According to the experimental results, the proposed method achieved the highest payload while maintaining the lowest distortion for most standard test images, comparing to other existing histogram-shifting-based reversible data hiding techniques.

## 1. Introduction

Data hiding is a technique that embeds data into cover media by slightly modifying their content [1] and has been used in many applications, such as tamper detection [2], copyright protection [3], and finger printing [4]. When data are embedded into cover media, the content of the media will be inevitably modified and thus distortion introduced. The distortion caused by data embedding is termed embedding distortion [5]. Although the embedding distortion in many applications is small, the distorted cover media cannot be recovered to their original state [6, 7]. However, some applications, such as in medical or military usages, allow no permanent embedding distortion in order to preserve content fidelity. This demand has highlighted the needs of reversible data hiding and has drawn much attention in the recent years [8–10].

The reversible data hiding is a technique that allows extraction of embedded data from the stego media and exactly restores the marked media to their original states [11]. Many researchers use digital image as the cover media because they are often transmitted throughout the Internet, which is easy to be accessed and may arouse a

little suspicious. An image that is used to embed data is called a cover image, and an image with data embedded is called a stego image [12]. The earliest reversible data hiding technique reported in the literature is Barton's work [13]. Afterwards, a number of reversible data hiding techniques have been proposed to fulfill the insatiate demands in this field. In 2003, Tian [14] proposed a novel reversible data hiding method with high payload. In his method, the difference value between paired pixels is expanded, and a bit can be embedded into the LSB of the expanded difference. In Tian's method,  $n$  bits can be embedded into  $2n$  pixels. Alattar [15] extended Tian's work by increasing the payload without introducing a noticeable distortion. In Alattar's method,  $n$  bits can be embedded into  $n + 1$  pixels. Tian and Alattar et al.'s embedding techniques can be classified as the expansion-embedding technique. For an expansion-embedding technique, difference values between pixels have to be expanded to conceal data. Therefore, the embedding distortion is relatively large. Besides, the selection of embedding position to avoid the overflow or underflow problem has to pay the overhead cost, which may significantly reduce the payload.

In 2006, Ni et al. [16] proposed a reversible data hiding method and achieved very high image quality. They selected pairs of peak and zero of an image histogram and shift the histogram bins to leave embeddable spaces for data embedding. Ni et al.'s embedding method can be classified as the shifting-embedding technique. In their work, the maximum payload is limited by the peak height of the cover image histogram; therefore, the payload is smaller compared to the expansion-embedding-based reversible data hiding techniques. In 2007, Thodi and Rodríguez combined the expansion-embedding and shifting-embedding techniques and proposed a reversible data hiding method with higher payload and lower distortion [5]. In their works, the prediction errors are expanded, and data are embedded into the LSBs of the expanded prediction errors. A better performance was achieved in Thodi et al.'s method than that of Tian's and Ni et al.'s methods.

Recently, some researchers [17–20] adopted the concept of shifting-embedding technique and embedded data into the prediction error histogram. Since the peak height of the prediction error histogram is usually higher than that of the image histogram for most natural images, a higher payload can be achieved. In these methods, the peak value of the prediction error histogram is calculated, and the histogram bin to the left or to the right of the peak value is shifted to vacate a histogram bin just next to the peak. Data is then embedded by modifying the prediction errors with the peak value. We classified these newly proposed methods as the prediction-and-shifting embedding (PSE) method because the data embedding of these methods mainly relies on prediction and histogram-shifting techniques. For a PSE method, the performance of the predictor plays a very important role. The peak height of a prediction error histogram should be as high as possible since the peak height represents the number of bits that can be embedded in a prediction error histogram. One goal of the PSE method is to construct a higher prediction error histogram to increase the payload. A higher prediction error histogram often results from an accurate prediction, which can be done by employing a well designed predictor that fully exploited the correlation among the neighboring pixels.

The PSE methods proposed so far have better embedding efficiency than the traditional histogram-shifting embedding technique; however, the existing methods offer no selective embedment mechanism (SEM) to exclude those pixels that contribute no capacity but merely cause image distortion. In this paper, a novel reversible data hiding method based on PSE is proposed. A sophisticated SEM is introduced to exclude pixels with larger prediction errors being modified to enhance the image quality. In comparison to prior reversible data hiding methods, the proposed method achieves the best performance in terms of payload and PSNR.

The rest of this paper is organized as follows. In Section 2, three PSE-based reversible data hiding methods will be described. The proposed method is presented in Section 3, followed by the experimental results and discussions in Section 4. Conclusions are addressed in Section 5.

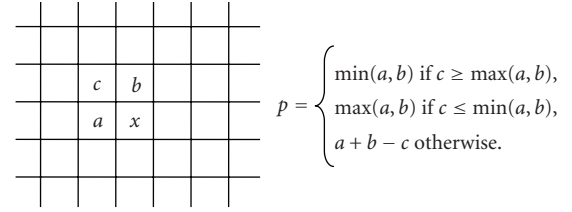


FIGURE 1: Context for predicting the pixel  $x$ .

## 2. Relative Works

In this section, we will briefly describe the central idea of PSE and will review three PSE-based reversible data hiding methods proposed in 2009.

**2.1. The Basic Concept of PSE.** The basic idea of the PSE method is to exploit the correlation of neighboring pixels inherent in images and predict the pixel values to obtain the prediction errors. To embed data, the prediction errors are scanned sequentially. If the scanned prediction error  $e$  is equal to some predetermined values (i.e., peak values), a bit  $s$  can be embedded by modifying  $e$  according to the value of  $s$ . Otherwise,  $e$  is modified to a value that will not cause ambiguous conduction when extracting data. The stego image can be constructed by transforming the modified prediction errors back to their spatial values.

To extract the embedded data, the stego image is sequentially scanned, and the modified prediction errors are obtained. If the scanned modified prediction error  $e'$  is equal to some pre-determined values (i.e., peak values), a bit 0 or 1 is then extracted. To recover the original image,  $e'$  is modified back to its original value,  $e$ , and then transforms  $e$  back to its spatial value. Since the PSE method only modifies the prediction error values slightly (usually plus or minus one unit), the stego image quality is often higher than that of expansion-embedding-based techniques. Besides, the payload of PSE method is larger than those histogram-shifting-based techniques because the peak of the prediction error histogram is often higher than that of the original image histogram.

**2.2. Hong Et al.'s Method.** Hong et al. [18] employed a median edge detection (MED) predictor used in JPEG-LS to sequentially predict pixel values and embed data. In their work, a pixel  $x$  is predicted by previously visited pixels  $a$ ,  $b$ , and  $c$ , as shown in Figure 1. The prediction error is calculated by  $e = x - p$ , where  $p$  is the prediction value of  $x$ .

Hong et al. recognized that the prediction error histogram is sharply distributed and centered at zero. Therefore, they choose  $e = 0$  and  $e = -1$  as the peak values and employ the PSE technique to embed data. However, the current pixel  $x$  is predicted by previously modified pixels; the prediction may become less accurate. A less accurate prediction will decrease the peak height of the prediction error histogram, leading to a decrease in payload.

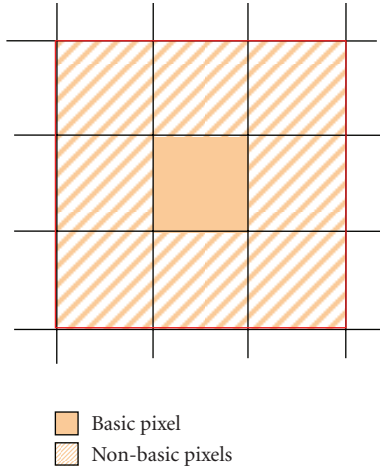


FIGURE 2: The layout of basic pixel and nonbasic pixel for a  $3 \times 3$  block.

**2.3. Tsai Et al.'s Method.** Tsai et al. [19] partitioned the cover image into blocks and the center pixel of each block is selected as the basic pixel of this block. The basic pixels served as reference pixels and will not be modified during the embedding process (i.e., the cover image and the stego image share the same set of basic pixels). The layout of basic pixel and nonbasic pixel for a  $3 \times 3$  block is shown in Figure 2.

To embed data, the value of the basic pixel is subtracted from other nonbasic pixels in the same block. The resulting difference values are the prediction errors of those nonbasic pixels. All blocks are processed in the same manner, and all the prediction errors can be obtained. After that, the PSE technique is employed to embed data. In Tsai et al.'s method, a larger payload might be achieved by decreasing the number of basic pixels or equivalently, increasing the block size. However, the predictor they used is simply the nearest predictor in that the nonbasic pixel value is predicted by the value of the nearest basic pixels. A larger block may result in a less accurate prediction, leading to a decrease in payload. In Tsai's method, a  $3 \times 3$  block is suggested to achieve the best results.

**2.4. Kim Et al.'s Method.** Kim et al. [20] exploited the spatial correlation between subsampled images and proposed a reversible data hiding method with high payload and low distortion. They subsampled the cover image into  $k$  subsampled images, and the subsampled image that maximizes the spatial correlation among the subsampled images is selected as the reference subimage. Figure 3 shows an example of a cover image with four subsampled images.

To obtain the prediction errors, the pixel values in the reference subimage are subtracted from other subsampled images. The resulting difference values are the prediction errors. The PSE technique is then applied for data concealment. Kim et al. also provided an adjustable embedding level mechanism for large payload at the cost of image distortion. In Kim et al.'s method, they simply use the value in the reference image as the prediction value of other subsampled

images, which is equivalent to adopt the nearest predictor and use the value in the reference image to predict pixel values in other subsampled images in the corresponding position. The use of the nearest predictor may result in a less accurate prediction [21] and subsequently reduce the payload.

### 3. Proposed Method

In this section, we present a PSE-based reversible data hiding method that achieves high payload with low distortion. In a PSE method, the embedding capacity is determined by the peak height, that is, the most frequently occurring value in the prediction errors. We term those errors "embeddable errors" because one bit can be embedded within one error. Prediction errors other than embeddable errors are termed "nonembeddable errors." During embedding, nonembeddable errors have to be shifted or remain unchanged according to the design of the embedding algorithm. Figure 4(a) shows the relationship between embeddable errors and nonembeddable errors, and Figure 4(b) illustrates a histogram of a one-side shifted embedding algorithm in which different types of prediction errors are marked accordingly.

For all prediction errors, only embeddable errors contribute to the payload. Those nonembeddable errors that have to be shifted during embedding contribute no payload but cause distortions. To enhance the embedding efficiency, it is desirable not only to increase the number of embeddable errors but also to decrease the number of those must-be-shifted nonembeddable errors.

It is known that the prediction errors of those pixels located in a complex region are often larger than those pixels located in a smooth region, and large prediction errors are likely to be nonembeddable and most of them have to be shifted. For example, Figure 5 shows an error image of Lena using an MED predictor. Note that different predictors may result in roughly the same error image since edges in the error image are mostly preserved. The vertical bar indicates the magnitude of the absolute prediction errors. Note that error values in complex regions (e.g., the hairs and edges) are larger than those in smooth regions (e.g., the shoulder). If a large prediction error can be detected before embedding, this prediction error can be excluded to join the embedding process and thus, the distortion can be reduced. A predictor used in a PSE method should have the capability to increase the height of the error histogram while reducing the number of nonembeddable errors that have to be shifted during embedding.

**3.1. The Selective Embedment Mechanism.** In this subsection, a selective embedment mechanism (SEM) is introduced. SEM employs a local smoothness estimator to determine whether the scanned pixel will be selected to go through the embedding process or just skipped. For each pixel  $I_{i,j}$ , a local smoothness estimator  $f_s(\cdot)$  is employed to estimate the smoothness for which the pixel  $I_{i,j}$  is located. The smoothness estimator  $f_s(I_{i,j})$  is defined as the standard

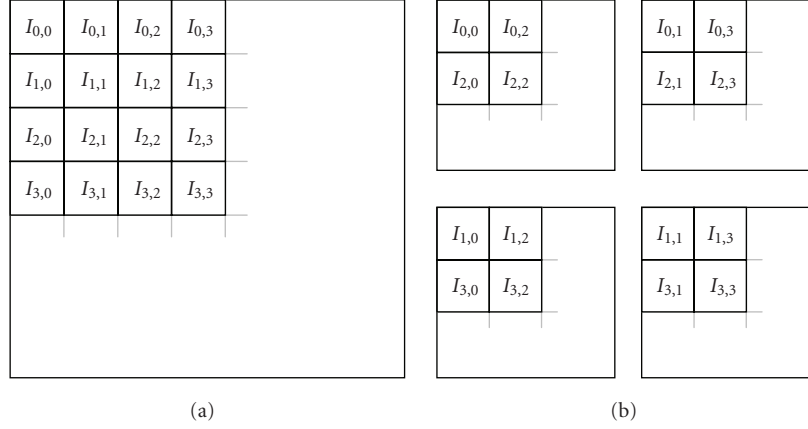


FIGURE 3: An example of image subsampling: (a) Original image and (b) Four subsampled images.

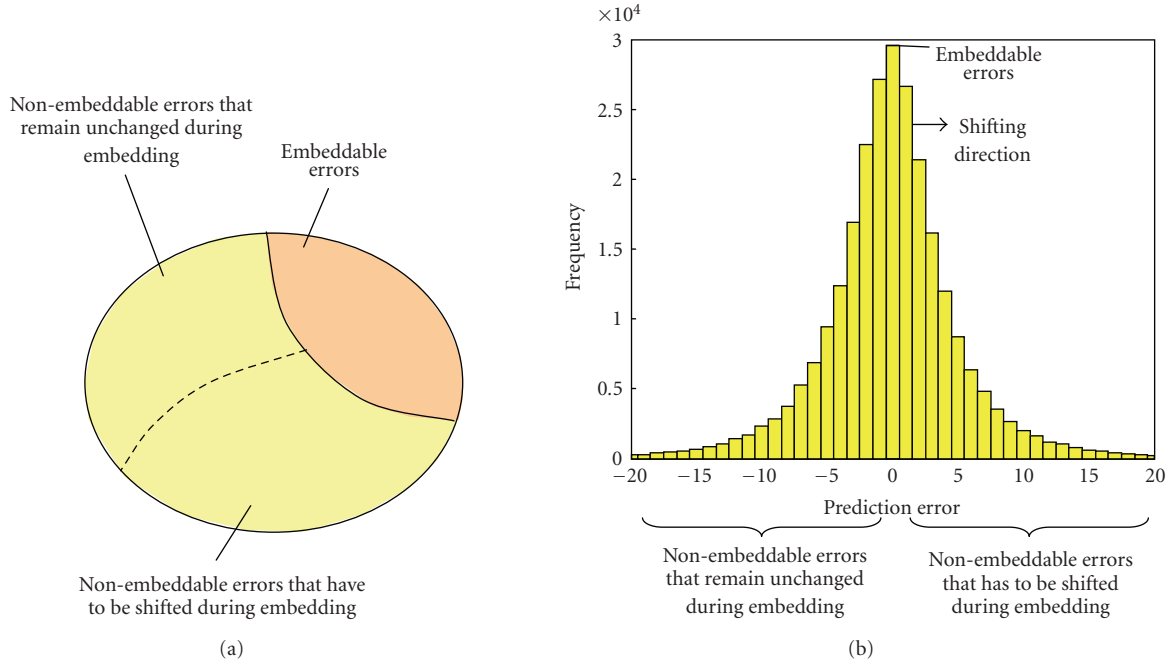


FIGURE 4: An illustration of embeddable and nonembeddable errors: (a) an illustration; (b) locations of embeddable and nonembeddable errors in a prediction error histogram.

deviation of the four neighboring pixels of the current pixel  $I_{i,j}$ :

$$f_s(I_{i,j}) = \sqrt{(I_{i,j-1} - \mu)^2 + (I_{i-1,j} - \mu)^2 + (I_{i,j+1} - \mu)^2 + (I_{i+1,j} - \mu)^2}, \quad (1)$$

where  $\mu$  is the mean of  $I_{i,j-1}$ ,  $I_{i-1,j}$ ,  $I_{i,j+1}$ , and  $I_{i+1,j}$ .

If  $f_s(I_{i,j}) < T$ , where  $T$  is a predefined threshold, then  $I_{i,j}$  is estimated in a smooth region. If  $f_s(I_{i,j}) \geq T$ ,  $I_{i,j}$  is estimated in a complex region. The prediction error of a pixel located in a complex region is likely to be large, and a larger prediction error is likely to be shifted but contributes no

payload; therefore, we simply skip  $I_{i,j}$  to reduce unnecessary pixel value modification. The SEM based on the threshold  $T$  is shown in Figure 6.

**3.2. The Prediction Algorithm.** The prediction algorithm of the proposed method is inspired by the error coding architecture used in multilevel progressive compression (MLP) method [22], where a linear weighted predictor  $P_{16}(\cdot)$  is employed to obtain prediction errors with a small variance. To begin with, pixels in the cover image are divided into two disjoint sets, namely, “Black” and “White” pixels in the same way as the layout of a checkerboard. Pixel  $I_{i,j}$  is “Black” if  $i + j$  is odd and is “White” otherwise. The embedding process involves two prediction passes. In the

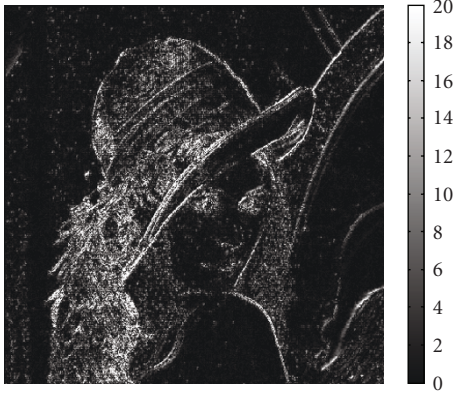


FIGURE 5: The image of prediction errors.

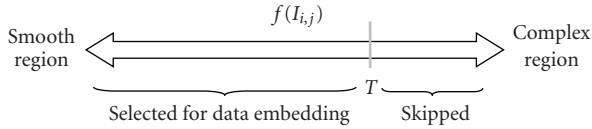


FIGURE 6: The selective embedment mechanism.

first pass, the values of all “Black” pixels  $I_{i,j}$  are predicted using the 16 known “White” neighbors. During embedding, the values of these “Black” pixels are modified. In the second pass, the “White” pixels are predicted using the 16 known, modified “Black” neighbors. Figure 7(a) shows the context of prediction neighborhood of the pixel to be predicted.

The prediction of  $I_{i,j}$  by using the predictor  $P_{16}(\cdot)$  is given by

$$\begin{aligned}
 P_{16}(I_{i,j}) = & 0.3164(I_{i,j-1} + I_{i-1,j} + I_{i,j+1} + I_{i+1,j}) \\
 & - 0.0351(I_{i-1,j-2} + I_{i-2,j-1} + I_{i-2,j+1} + I_{i-1,j+2} \\
 & \quad + I_{i+1,j+2} + I_{i+2,j+1} + I_{i+2,j-1} + I_{i+1,j-2}) \\
 & + 0.0039(I_{i,j-3} + I_{i-3,j} + I_{i,j+3} + I_{i+3,j}).
 \end{aligned} \quad (2)$$

The weights in  $P_{16}(\cdot)$  are calculated by bicubic polynomial interpolation and are normalized, so that their sum is one [22].

For pixels located near the border of the cover image and have no sufficient neighbors to predict, we may simply skip these pixels, or slightly modify the prediction rules, so that these pixels can be applied [22]. Because there is only a small portion of pixels that have no sufficient neighbors, these effects have little practical significance.

**3.3. The Embedding Algorithm.** To embed data, the local smoothness estimator is employed to exclude pixels located in complex regions to join the embedding process. The pixels located in smooth regions are then predicted, and

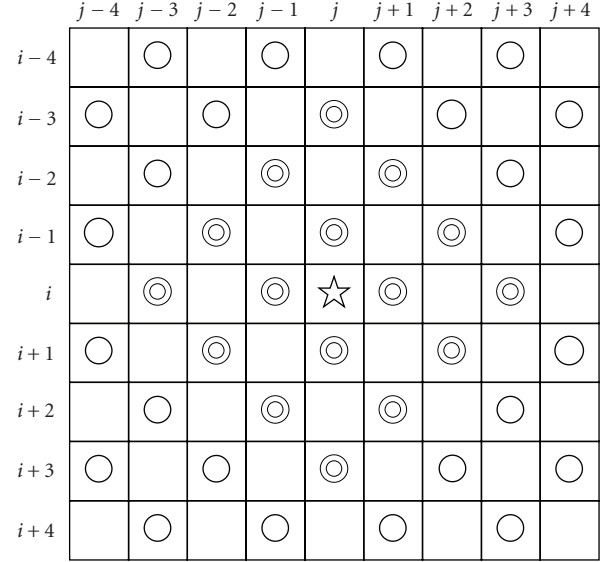


FIGURE 7: An illustration of the prediction algorithm: the pixel values at the position marked  $\odot$  and  $\circ$  are known, the pixels at the positions marked  $\odot$  are used to predict the value of center pixel (marked  $\star$ ), and all other unmarked pixels will be predicted in the second prediction pass.

data embedding is done by modifying the prediction errors. The detailed embedding procedure is listed in the following.

**Input.** A cover image  $I$  of size  $M \times M$  and secret data  $S$ .

**Output.** A stego image  $I'$ , a minimum threshold  $T_{\min}$ , end of embedding position  $E_p$ , two pairs of peaks  $(p_0^+, p_0^-)$ ,  $(p_1^+, p_1^-)$ , and a location map  $L_M$ .

**Step 1.** Set  $k = 0$ ,  $T = 0$ .

**Step 2.** Scan the pixels in  $I$  using the raster scan order. For pixels  $I_{i,j}$ s satisfying  $\text{mod}(i+j, 2) = k$ , where  $\text{mod}(x, 2)$  is a modulus function that returns 1 if  $x$  is an odd number and 0 otherwise, the estimator  $f_s(\cdot)$  is employed to estimate the local smoothness of those pixels. If  $f_s(I_{i,j}) \leq T$ , pixel  $I_{i,j}$  is classified to be within a smooth region. Thus, the predictor  $P_{16}(\cdot)$  is employed to calculate the prediction value  $\hat{I}_{i,j}$  of  $I_{i,j}$ . The prediction error is then calculated by  $e_{i,j} = I_{i,j} - \hat{I}_{i,j}$ . If  $f_s(I_{i,j}) > T$ , the scanned pixel is classified to be within a complex region. Therefore, the prediction error will not be calculated, that is, the scanned pixel will not join the embedding process.

**Step 3.** After all the prediction errors are obtained, the histogram of the prediction errors is calculated, and a pair of peaks  $(p_k^+, p_k^-)$ , where  $p_k^+ > p_k^-$ , of the histogram is determined.



*Step 4.* Scan each prediction error  $e_{i,j}$  obtained in Step 2. If the scanned error  $e_{i,j}$  is equal to  $p_k^+$  or  $p_k^-$ , the scanned error is embeddable, and a bit  $s$  can be embedded by using the rules

$$e'_{i,j} = \begin{cases} e_{i,j} + 1 & \text{if } e_{i,j} = p_k^+, s = 1, \\ e_{i,j} - 1 & \text{if } e_{i,j} = p_k^-, s = 1, \\ e_{i,j} & \text{if } (e_{i,j} = p_k^+ \text{ or } e_{i,j} = p_k^-), s = 0. \end{cases} \quad (3)$$

Otherwise, the scanned error  $e_{i,j}$  is nonembeddable and has to be modified to  $e'_{i,j}$  by using the rules

$$e'_{i,j} = \begin{cases} e_{i,j} + 1 & \text{if } e_{i,j} > p_k^+, \\ e_{i,j} - 1 & \text{if } e_{i,j} < p_k^-, \\ e_{i,j} & \text{otherwise.} \end{cases} \quad (4)$$

The pixel value  $I_{i,j}$  in the cover image is then modified to  $I'_{i,j}$  by setting  $I'_{i,j} = \hat{I}_{i,j} + e'_{i,j}$ . If  $I'_{i,j} > 255$  or  $I'_{i,j} < 0$ , an overflow or underflow problem occurs. If this happened, record the position information  $(i, j)$  in the location map  $L_M$ , mark  $s$  unextracted if  $s$  had been extracted from  $S$ , and set  $e'_{i,j} = e_{i,j}$ .

*Step 5.* Set  $k = 1$  and repeat Steps 2–4 to obtain another pair of peaks  $(p_k^+, p_k^-)$  and perform data embedding.

*Step 6.* Repeat Steps 1–5 and perform a binary search to find a minimum threshold  $T_{\min}$  such that  $|p_0^+| + |p_0^-| + |p_1^+| + |p_1^-|$  is just larger than the length of  $S$ , where  $|x|$  represents the number of bits that is used to record  $x$ . The end of embedding position  $E_P$  is recorded for the purpose of data extraction.

*Step 7.* Output the stego image  $I'$ , and the parameters  $T_{\min}$ ,  $p_0^+$ ,  $p_0^-$ ,  $p_1^+$ ,  $p_1^-$ ,  $E_P$ , and  $L_M$  are served as a key  $K$  for decoding.

The key  $K$  is transmitted over a secret channel. The receiver with the correct key  $K$  can then extract the embedded message and restore the stego image to the original image.

We use eight bits to record each of  $T_{\min}$ ,  $p_0^+$ ,  $p_0^-$ ,  $p_1^+$ , and  $p_1^-$ , use  $\lceil \log_2(M \times M) \rceil$  bits to record  $E_P$ , and use  $N \times \lceil \log_2(M \times M) \rceil$  bits to record  $L_M$  if there are  $N$  overflow and underflow pixels. The proposed embedding algorithm only modified the pixel values plus or minus one grayscale unit at most, and pixel values at 0 or 255 occur rarely for most natural images. In this case, for a  $512 \times 512$  cover image with no pixel value at 0 or 255, that is, no overflow and underflow occurs, the key size  $|K|$  is  $5 \times 8 + \lceil \log_2(512 \times 512) \rceil = 58$  bits.

**3.4. The Extraction and Recovery Procedures.** Once the receiver receives the stego image  $I'$  and the key  $K$ , the embedded secret data can be extracted, and the original image can be recovered by using the procedure listed below.

Input. The stego image  $I'$  and the key  $K$ .

Output. Recovered cover image  $I$  and the secret data  $S$ .

*Step 1.* Set  $k = 1$ .

*Step 2.* Scan the pixels  $I'_{i,j}$ 's with their positions satisfying  $\text{mod}(i + j, 2) = k$  using the same order as in the embedding phase. If the position of scanned pixel is recorded in  $L_M$ , this pixel is skipped and proceeds to next one. If  $f_s(I_{i,j}) \leq T_{\min}$ , then the prediction value  $\hat{I}'_{i,j}$  of  $I'_{i,j}$  is calculated by using the predictor  $P_{16}(\cdot)$ , and the prediction error can be obtained by  $e'_{i,j} = I'_{i,j} - \hat{I}'_{i,j}$ .

*Step 3.* If  $e'_{i,j} = p_k^+$  or  $e'_{i,j} = p_k^-$ , a bit  $s$  can be extracted by using the following rule:

$$s = \begin{cases} 0 & \text{if } e'_{i,j} = p_k^+ \text{ or } e'_{i,j} = p_k^-, \\ 1 & \text{if } e'_{i,j} = p_k^+ + 1 \text{ or } e'_{i,j} = p_k^- - 1, \end{cases} \quad (5)$$

The original prediction error  $e_{i,j}$  can be recovered by using the following rule:

$$s = \begin{cases} e'_{i,j} - 1 & \text{if } e'_{i,j} > p_k^+, \\ e'_{i,j} + 1 & \text{if } e'_{i,j} < p_k^-, \\ e'_{i,j} & \text{otherwise.} \end{cases} \quad (6)$$

The original pixel value  $I_{i,j}$  can be obtained by calculating  $I_{i,j} = \hat{I}_{i,j} + e_{i,j}$ .

*Step 4.* Repeat Steps 2–3 until the end of embedding position  $E_P$  is met.

*Step 5.* Set  $k = 0$  and repeat Steps 2–4. Concatenating the extracted bits, the embedded secret data can be obtained.

## 4. Experimental Results and Discussions

Several experiments, including tests on 8-bit and 16-bit images as well as using steganalysis tools, were carried out to demonstrate the effectiveness of the embedding algorithms proposed in this paper.

**4.1. 8-Bit Test Images.** Six standard  $512 \times 512$  images Airplane, Lena, Sailboat, Peppers, Boat, and Baboon taken from USC-SIPI database [23] were converted to 8-bit grayscale image by using the following equation if they were originally in RGB color format:

$$V = 0.2989R + 0.5870G + 0.1140B, \quad (7)$$

where  $V$  is the converted grayscale value and  $R$ ,  $G$ , and  $B$  are the red, green, and blue component of the cover image. The six grayscale test images are shown in Figure 8. The secret data were generated by using a pseudorandom number generator (PRNG). The peak signal-to noise ratio (PSNR) is used to measure the stego image quality:

$$\text{PSNR} = 10 \log_{10} \left( \frac{(2^b - 1)^2}{\text{MSE}} \right), \quad (8)$$



FIGURE 8: Six standard grayscale images.

where  $b$  is the bit depth for the cover image and MSE is the mean square error between the cover image and the stego image. The payload  $\rho$  is measured in bpp and is calculated by

$$\rho = E_C - O, \quad (9)$$

where  $E_C$  denotes the capacity of the given cover image and  $O$  denotes the side information that is required at the decoding stage. In the proposed method, the size of  $O$  is equal to the size of the key  $|K|$ .

According to our experiments, no overflow or underflow occurred in the test images Airplane, Lena, Sailboat, and Baboon. The key size for these four images is 58 bits. On the other hand, there is one pixel and eight pixels overflow or underflow for Peppers and Boat, respectively. The key size is  $58 + 1 \times \log_2(512 \times 512) = 76$  bits for Peppers and  $58 + 9 \times \log_2(512 \times 512) = 202$  bits for Boat.

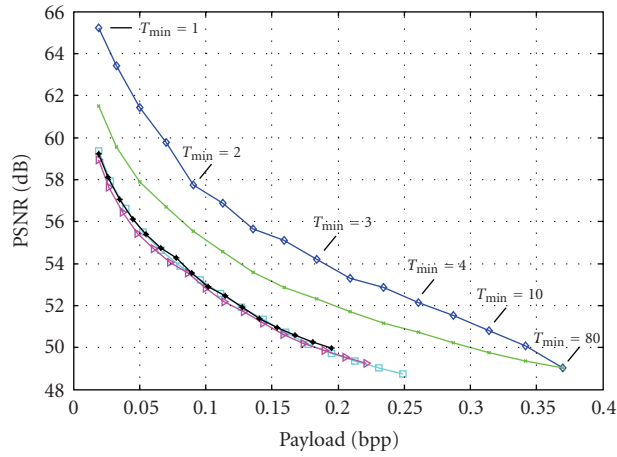
To compare the proposed method with Hong et al.'s, Tsai et al.'s and Kim et al.'s methods proposed in 2009, the embedding algorithms of these methods were implemented, and the parameters for each method were chosen, so that the best performance can be achieved. For Hong et al.'s method, two peaks 0 and  $-1$  were selected for embedding, as suggested in their paper. For Tsai et al.'s method, a  $3 \times 3$  block size was employed since the best result can be achieved

TABLE 1: Key size comparison.

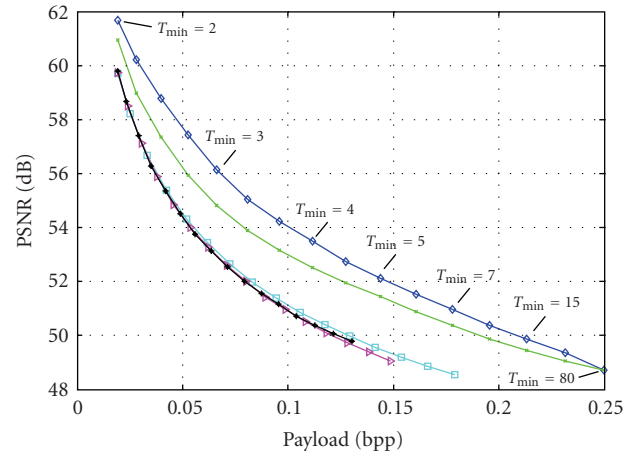
Method	Key size (bits)
Proposed method	$58 + 18N$
Hong et al.	$8 + 18N$
Tsai et al.	$32 + 18N$
Kim et al.	$10 + 18N$

by using blocks of this size. For Kim et al.'s method, four subsampled images were used, and the histogram bins were shifted one unit at most to ensure that a high-quality stego image can be achieved.

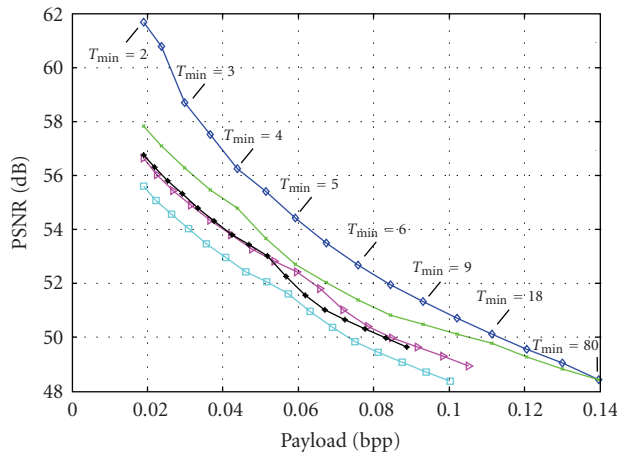
In Hong et al.'s method, the key requires eight bits to record the end of embedding position. In Tsai et al.'s method, the key is composed of two pairs of peak and zero points; each pair requires 16 bits to record it. In Kim et al.'s method, the key is composed of two sampling factors, three bits each, and four bits embedding level. To ensure that the pixels-to-be-modified are changed one grayscale unit at most to achieve high stego image quality, these methods also require a location map to prevent overflow or underflow occurrence. Suppose that there are  $N$  overflow and underflow pixels for a  $512 \times 512$  cover image, the key size of each method is listed in Table 1.



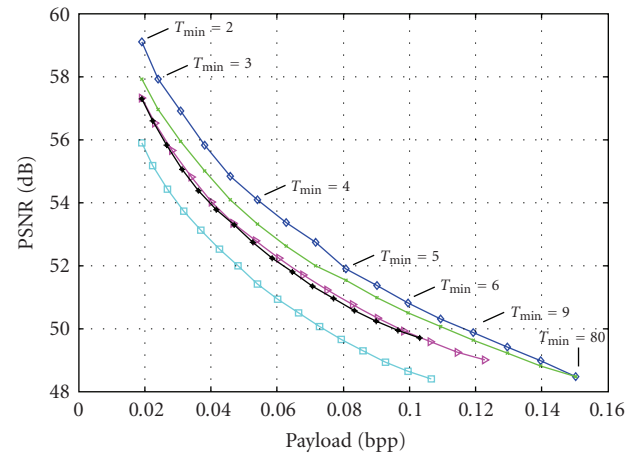
(a) Airplane



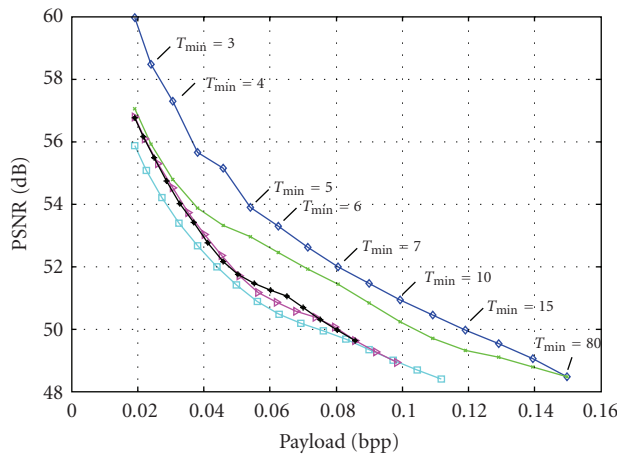
(b) Lena



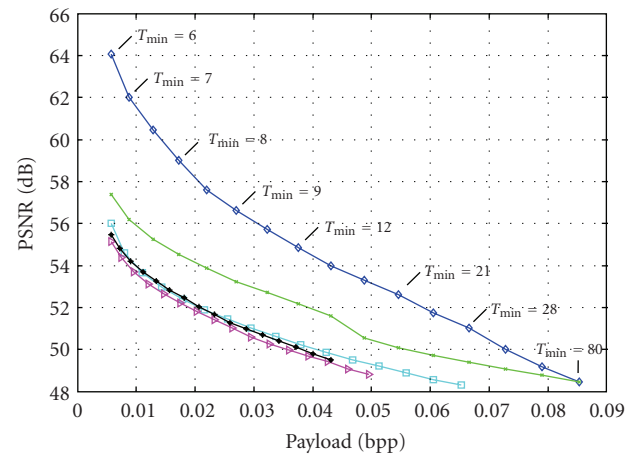
(c) Sailboat



(d) Peppers



(e) Boat



(f) Baboon

—◆— Proposed method with SEM      —▲— Tsai et al.'s method  
 —■— Proposed method without SEM      —●— Kim et al.'s method  
 —□— Hong et al.'s method

—◆— Proposed method with SEM      —▲— Tsai et al.'s method  
 —■— Proposed method without SEM      —●— Kim et al.'s method  
 —□— Hong et al.'s method

FIGURE 9: A comparison of various test images.



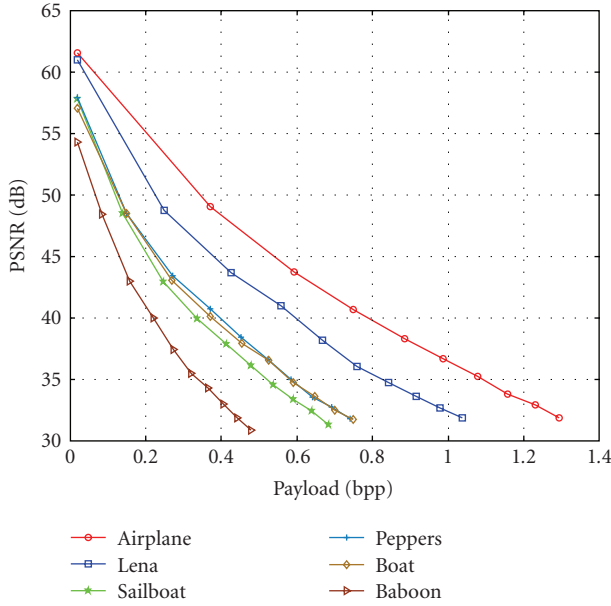


FIGURE 10: Comparison of payload versus PSNR for test images.

For the PSE-based method, the overflow and underflow problems will only occur at pixels of the cover image valued 0 or 255. This occurs rarely for most natural images. Therefore, the key size listed in Table 1 is comparable and is suitable to be transmitted over the Internet.

The comparison results of bpp-PSNR relationship for different methods are shown in Figures 9(a)–9(f). The minimum thresholds  $T_{\min}$  are marked beside the dots on the curve for the proposed method. Note that the dots that are not marked represent having the same previous  $T_{\min}$ . The lower bond of PSNR of these 8-bit test images is  $10 \times \log_{10}(255^2/1) \approx 48.13$  dB because pixels in the cover images are modified one grayscale unit at most, resulting in that the MSE between the cover image and the stego image is slightly smaller than 1.

In the proposed method, the advantage of using SEM can be seen from Figures 9(a)–9(f). For example, Figure 9(a) reveals that, when SEM is applied, the gain in PSNR at 0.1 bpp is around 2 dB for the smooth image Airplane. At this moment, setting  $T_{\min} = 1$  is enough to embed all secret data. The threshold  $T_{\min}$  is gradually increased as the embedding rates increased. This is because in SEM, a larger threshold will be selected in order to embed more data; however, the advantage of using SEM to increase the PSNR will become less significant as the payload increases. It is clear that the proposed method with SEM performs the best for all the test images at all embedding rates than those without SEM. This is because SEM evaluates a minimum threshold to prevent pixels located in complex regions from being selected for data embedding, since these pixels often contribute fewer payloads but cause almost equally distortion.

For the complex image Baboon, the gain in PSNR by using SEM becomes significantly larger when the embedding rates are small, as shown in Figure 9(f). This is because a smaller threshold  $T_{\min}$  is used to select those pixels located

in smooth regions for data embedding, and these pixels often contribute more embeddable errors. On the other hand, if no SEM is applied, pixels located in smooth or complex regions will have equal probability to be selected for data embedding, resulting in significant image degradation. Figures 9(a)–9(f) also reveal that the proposed method with SEM performs better than those without using this mechanism at all embedding rates, and the improvements are significant from small to moderate payload.

Note that the proposed method outperforms Hong et al.'s, Tsai et al.'s, and Kim et al.'s methods in terms of payload and PSNR for all test images, at all embedding rates, even if no SEM is used. For example, for the smooth image, such as Lena, the proposed method achieved 54 dB at 0.1 bpp whereas their methods only achieved around 51 dB at the same bit rate. For the complex image, such as Baboon, the proposed method also performs much better. For example, the PSNR of the proposed method is 54.5 dB at 0.04 bpp whereas the PSNR of their method only achieved around 50 dB under the same bit rate.

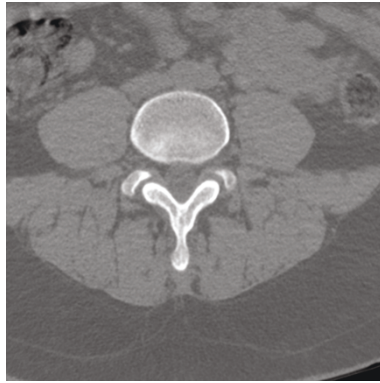
Although the proposed method focused on high-quality stego images, it can be implemented for large payload using a multi-level embedding strategy, namely, the stego image is the cover image for the next embedding level. In this case, the proposed method without SEM version is used to speed up the embedding process because the contribution of SEM becomes insignificant at high payload. The side information produced in each embedding level is embedded together with the data bits into the next embedding level; only the side information produced in the last embedding level is served as the key  $K$ . In the decoding phase, the key for the previous embedding level and data embedded in the last embedding level are extracted with the key  $K$ , and the stego image is restored to its previous state. This process is repeated until all the data bits are extracted and the original image is recovered. Figure 10 shows the payload versus PSNR for each test images with 10 embedding levels. As shown in the figure, the quality of the stego image is high at low and moderate payloads and is still acceptable at high payloads. Since the proposed method is based on PSE technique, the payload-distortion performance depends on the characteristics of the cover image. A better performance, that is, a higher payload with a lower distortion, can be achieved when the cover image contains large amount of smooth regions, for example, Airplane and Lena. On the contrary, cover images with large amount of complex regions, such as Sailboat and Baboon, often exhibit lower PSNR under the same payload.

We also tested the proposed method and Hong et al.'s, Tsai et al.'s, and Kim et al.'s methods using 23 natural photographic images of the Kodak images test set, each sized  $768 \times 512$ . These images were also used in Hong et al.'s experiments [18]. The results were shown in Table 2 and have significant improvement in payload under roughly the same PSNR. Note that the averaged payload is one-third higher than their methods under roughly the same PSNR.

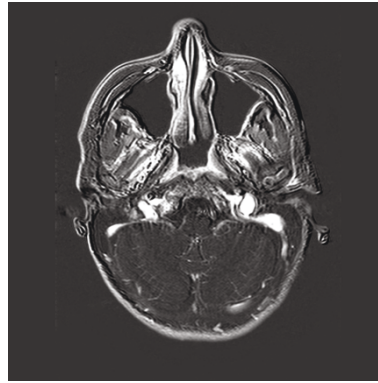
It is interesting to note that for Image no. 8, the proposed method provides fewer payloads than that of Hong et al.'s method. This is because in this particular image, rich vertical and horizontal edges provide the MED predictor a better

TABLE 2: Maximum payload for various test images (payload is measured in bits).

Image	Hong et al.		Tsai et al.		Kim et al.		Proposed	
	Payload	PSNR	Payload	PSNR	Payload	PSNR	Payload	PSNR
1	52,743	48.45	37,951	48.96	41,098	48.92	58,692	49.35
2	90,417	48.68	80,074	49.26	83,002	49.21	123,689	48.95
3	114,193	48.83	111,829	49.49	115,578	49.45	162,458	49.27
4	84,229	48.64	69,653	49.18	72,595	49.14	118,052	48.95
5	59,630	48.49	41,324	48.99	44,615	48.95	78,698	49.75
6	73,067	48.57	66,355	49.16	69,220	49.12	93,673	49.23
7	107,538	48.78	94,483	49.36	97,741	49.32	155,529	49.44
8	53,113	48.45	34,399	48.94	37,507	48.9	43,523	50.29
9	82,651	48.63	77,472	49.24	80,802	49.2	106,925	49.05
10	83,997	48.64	74,045	49.21	77,538	49.17	109,604	49.02
11	79,063	48.61	67,484	49.17	70,677	49.13	100,481	49.16
12	96,509	48.71	86,945	49.3	89,766	49.26	127,128	49.09
13	32,631	48.33	24,386	48.87	27,222	48.83	38,553	49.56
14	59,988	48.49	42,461	48.99	46,184	48.96	78,118	49.02
15	98,337	48.72	98,950	49.39	101,497	49.34	133,588	49.28
16	88,007	48.66	76,246	49.23	79,415	49.19	118,570	49.07
17	82,961	48.63	71,970	49.2	75,200	49.16	113,159	48.81
18	49,885	48.43	40,651	48.98	40,698	48.92	67,991	48.99
19	70,976	48.56	59,135	49.11	60,320	49.05	86,501	49.36
20	127,419	48.91	155,474	49.83	156,857	49.77	143,470	50.56
21	69,064	48.54	63,928	49.14	64,509	49.08	87,292	49.22
22	69,066	48.54	52,931	49.06	53,853	49.01	95,492	48.9
23	107,315	48.78	102,726	49.42	104,379	49.37	151,973	49.25
Avg.	79,686	48.61	70,907	49.19	73,490	49.15	104,050	49.29



(a) im1



(b) im2

FIGURE 11: Two medical images.

opportunity to produce higher prediction error histogram than the proposed MLP predictors. Nevertheless, the PSNR of the proposed method is 1.35 dB higher than Hong et al.'s method.

**4.2. 16-Bit Test Images.** Since the proposed method particularly focuses on applications where high-quality stego image is demanded such as medical images, the proposed method was tested to two 16-bit  $512 \times 512$  medical images

obtained from [24, 25] as shown in Figure 11. The medical images, im1 and im2 shown in Figure 11, are indeed two very different images in their contents. Most regions in im1 are informative parts containing body tissues whereas in im2, the informative parts are surrounded by almost uniform dark regions.

The proposed method was tested and compared with those of Tsai et al.'s, Kim et al.'s, Hong et al.'s, and a newly method proposed by Fallahpour et al. [26], which is primarily designed for medical images. Fallahpour et al.'s method

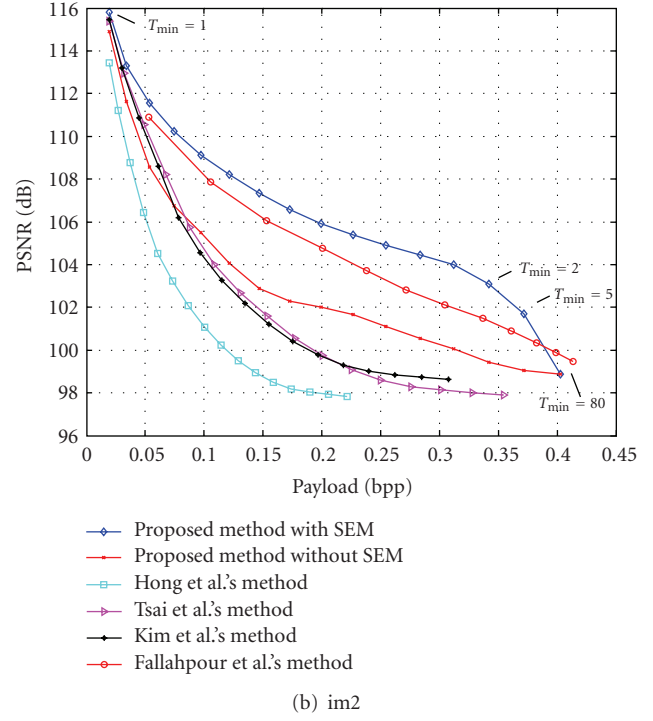
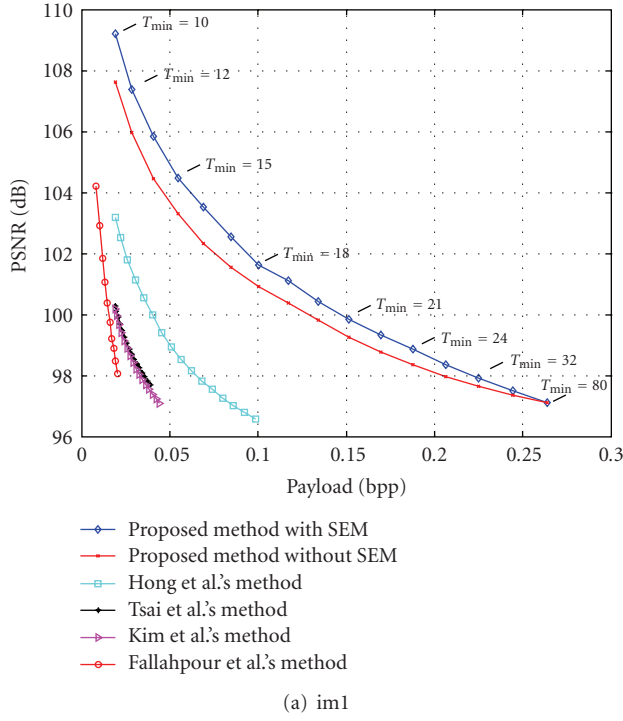


FIGURE 12: PSNR versus payload of two medical images.

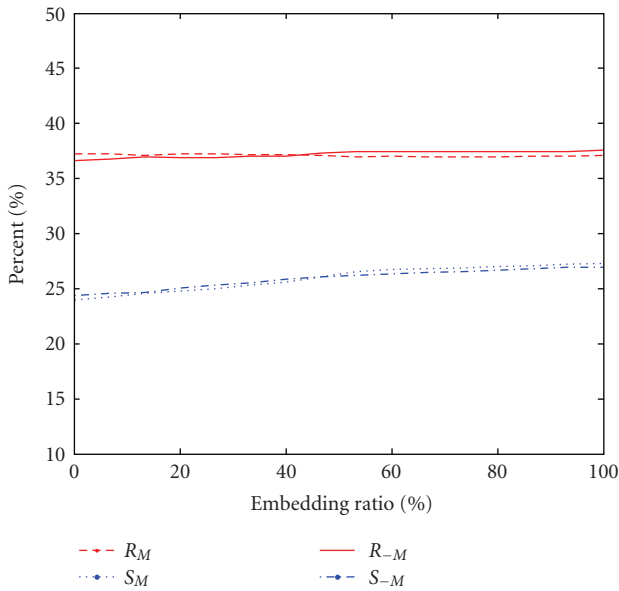


FIGURE 13: RS-diagram of Lena stego image.

is based on partitioning the image into nonoverlapping blocks, and these blocks are prioritized based on objective or subjective quality. Data are then embedded into each block at the pixel level using histogram-shifting technique. The results obtained from each method are shown in Figure 12. In our experiments, 16 image blocks were used in Fallahpour et al.'s method. Note that the lower bond of PSNR for 16-bit images is  $10 \log_{10}((2^{16} - 1)^2/1) \approx 96.33$  dB.

As shown in Figure 12, the proposed method outperforms other methods under various embedding rates. For example, the maximum payload of im1 of the proposed method is around 0.27 bpp whereas others are less than 0.1 bpp. For image im2, the performance of the proposed method is comparable to that of Fallahpour et al.'s method but outperforms other methods at all embedding rates. It is interesting to note that Fallahpour et al.'s method performs better in im2 than in im1. This is because the large uniform background in im2 offers sharply distribution image histograms and provides more embeddable spaces. On the contrary, the histograms of image blocks in im1 are relatively flat since the pixel intensities of image blocks are spread out, causing the embeddable spaces to significantly decreased.

**4.3. Security Verification.** Although the proposed method produces imperceptible high-quality stego image, there exist steganalysis tools to detect whether an image is embedded with messages. The RS-method proposed by Fridrich et al. [27] is one of the well-known steganalysis tools used to examine the security of a data hiding technique. The RS-method successfully detects the LSB embedding by using sensitive dual statistics derived from the information of the regular and singular (RS) grouping in images. To detect an image, the image is partitioned into groups  $G$  of  $n$  consecutive pixels. The discrimination function, the flipping function, and the mask  $M$  are used to classify the groups  $G$  into three disjoint groups: regular, singular, and unusable groups. The RS method analyzes the percentage of regular groups  $R_M$ ,  $R_{-M}$  and the percentage of singular groups  $S_M$ ,

$S_{-M}$ . For most natural images, the relationships  $R_M \simeq R_{-M}$  and  $S_M \simeq S_{-M}$  generally hold. If this relationship is violated, the embedded message is suspicious to be detected; see [27] for more details. The RS-diagram of the proposed method for the test image Lena is shown in Figure 13.

As can be seen in Figure 13, the relationships  $R_M \simeq R_{-M}$  and  $S_M \simeq S_{-M}$  hold for various embedding ratios. According to our experiments, other test images have similar RS-diagrams, indicating that the proposed method is secure from the RS-diagram steganalysis.

## 5. Conclusions

In this paper, we have presented a reversible data hiding scheme based on prediction-and-shifting embedding technique and achieved high payload and high image quality. The proposed method employs the SEM for determining the best threshold to exclude pixels located in complex regions to join the embedding process, so that the number of modified pixels can be greatly reduced. When large payload is embedded, multi-level embedding technique is performed. The proposed method has the following advantages: (1) simple and effective, (2) applicable to variety of images such as photographic or medical images, and (3) adjustable payload according to the requirement of applications. Test results showed that, for a variety of test images, the proposed method outperforms prior works, such as Hong et al.'s, Tsai et al.'s, Kim et al.'s, Thodi et al.'s, and Tian's methods in terms of payload and PSNR.

## Acknowledgment

This research was supported by the National Science Council of the Republic of China under Grant NSC98-2622-E-412-003-CC3.

## References

- [1] N. Provos and P. Honeyman, "Hide and seek: an introduction to steganography," *IEEE Security and Privacy*, vol. 1, no. 3, pp. 32–44, 2003.
- [2] P. L. Lin, C.-K. Hsieh, and P.-W. Huang, "A hierarchical digital watermarking method for image tamper detection and recovery," *Pattern Recognition*, vol. 38, no. 12, pp. 2519–2529, 2005.
- [3] S. Lee, C. D. Yoo, and T. Kalker, "Reversible image watermarking based on integer-to-integer wavelet transform," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 321–330, 2007.
- [4] A. K. Jain and U. Uludag, "Hiding biometric data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 11, pp. 1494–1498, 2003.
- [5] D. M. Thodi and J. J. Rodríguez, "Expansion embedding techniques for reversible watermarking," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 721–730, 2007.
- [6] J. Mielikainen, "LSB matching revisited," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 285–287, 2006.
- [7] Z. H. Wang, T. D. Kieu, C. C. Chang, and M. C. Li, "A novel information concealing method based on exploiting modification direction," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 1, no. 1, pp. 1–9, 2010.
- [8] M. Kuribayashi, M. Morii, and H. Tanaka, "Reversible watermark with large capacity based on the prediction error expansion," *Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E91-A, no. 7, pp. 1780–1790, 2008.
- [9] D. Coltuc and J.-M. Chassery, "Very fast watermarking by reversible contrast mapping," *IEEE Signal Processing Letters*, vol. 14, no. 4, pp. 255–258, 2007.
- [10] S. Han, M. Fujiyoshi, and H. Kiya, "An efficient reversible image authentication method," *Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E91-A, pp. 1907–1914, 2008.
- [11] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Lossless generalized-LSB data embedding," *IEEE Transactions on Image Processing*, vol. 14, no. 2, pp. 253–266, 2005.
- [12] J. Fridrich and D. Soukal, "Matrix embedding for large payloads," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 3, pp. 390–395, 2006.
- [13] J. M. Barton, "Method and apparatus for embedding authentication information within digital data," US patent 5 646 997, July, 1997.
- [14] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890–896, 2003.
- [15] A. M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Transactions on Image Processing*, vol. 13, no. 8, pp. 1147–1156, 2004.
- [16] Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 3, pp. 354–361, 2006.
- [17] M. Chen, Z. Chen, X. Zeng, and Z. Xiong, "Reversible data hiding using additive prediction-error expansion," in *Proceedings of the 11th ACM Workshop on Multimedia and security*, pp. 19–24, 2009.
- [18] W. Hong, T. S. Chen, and C. W. Shiu, "Reversible data hiding for high quality images using modification of prediction errors," *The Journal of Systems and Software*, vol. 82, no. 11, pp. 1833–1842, 2009.
- [19] P. Tsai, Y.-C. Hu, and H.-L. Yeh, "Reversible image hiding scheme using predictive coding and histogram shifting," *Signal Processing*, vol. 89, no. 6, pp. 1129–1143, 2009.
- [20] K. Kim, M. Lee, H.-Y. Lee, and H.-K. Lee, "Reversible data hiding exploiting spatial correlation between sub-sampled images," *Pattern Recognition*, vol. 42, no. 11, pp. 3083–3096, 2009.
- [21] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice-Hall, Upper Saddle River, NJ, USA, 2nd edition, 2002.
- [22] D. Salomon, *Data Compression: The Complete Reference*, Springer, Berlin, Germany, 2nd edition, 2000.
- [23] USC-SIPI Image Database, <http://sipi.usc.edu/database>.
- [24] Medical Image Samples, <http://www.barre.nom.fr/medical/samples>.
- [25] DICOM Sample Image Sets, <http://pubimage.hcuge.ch:8080>.
- [26] M. Fallahpour, D. Megias, and M. Ghanbari, "High capacity, reversible data hiding in medical images," in *Proceedings of the IEEE International Conference on Image Processing*, pp. 4241–4244, 2009.
- [27] J. Fridrich, M. Goljan, and R. Du, "Reliable Detection of LSB Steganography in Color and Grayscale Images," in *Proceedings of the International Workshop on Multimedia and Security*, pp. 27–30, 2001.