High-Quality and Robust Reversible Data Hiding by Coefficient Shifting Algorithm

Ching-Yu Yang and Chih-Hung Lin

This study presents two reversible data hiding schemes based on the coefficient shifting (CS) algorithm. The first scheme uses the CS algorithm with a mean predictor in the spatial domain to provide a large payload while minimizing distortion. To guard against manipulations, the second scheme uses a robust version of the CS algorithm with feature embedding implemented in the integer wavelet transform domain. Simulations demonstrate that both the payload and peak signal-to-noise ratio generated by the CS algorithm with a mean predictor are better than those generated by existing techniques. In addition, the marked images generated by the variant of the CS algorithm are robust to various manipulations created by JPEG2000 compression, JPEG compression, noise additions, (edge) sharpening, low-pass filtering, bit truncation, brightness, contrast, (color) quantization, winding, zigzag and poster edge distortion, and inversion.

Keywords: High-quality reversible data hiding, robust reversible data hiding, coefficients shifting, watermarking.

Manuscript received May 23, 2011; revised Oct. 3, 2011; accepted Nov. 18, 2011.

Ching-Yu Yang (phone: +886 9 2155 0648, chingyu@npu.edu.tw) is with the Department of Computer Science and Information Engineering, National Penghu University of Science and Technology, Penghu, Taiwan.

Chih-Hung Lin (chuck@mail.ncyu.edu.tw) is with the Graduate Institute of Mathematics and Science Education, National Chiayi University, Chiayi, Taiwan.

http://dx.doi.org/10.4218/etrij.12.0111.0312

I. Introduction

Reversible data hiding, also known as lossless data hiding, has been extensively studied during the last decade. A major difference between reversible data hiding and conventional data hiding/watermarking techniques [1]-[6] is that the former preserves the originality of valuable (or priceless) host media, including medical images, military maps, and geographic information. Generally speaking, the resulting perceived quality or peak signal-to-noise ratio (PSNR) and payload or bits per pixel (bpp) are the two most commonly-used criteria for evaluating the performance of reversible data hiding techniques. However, these criteria conflict with each other. Specifically, a reversible data hiding method with a quality perceived to be high often provides low capacity. Furthermore, most reversible data hiding schemes [7]-[14] are fragile, which means that hidden messages cannot be successfully extracted and the host media cannot be fully recovered when even the slightest alteration has been made to the marked images. Thus, the researchers have developed robust reversible data hiding schemes to solve the issue. A remarkable feature of a robust reversible data hiding scheme is that it can recover the host images completely if the marked images remain intact and restore (most part of) the hidden messages if the marked images have suffered from manipulations.

To obtain a desirable perceived quality with a high payload, this study proposes a reversible data hiding method that uses the coefficient shifting (CS) algorithm with a mean predictor. Subsequently, this study presents a robust reversible data hiding method using a variant of the CS algorithm that is based on the integer wavelet transform (IWT) domain to resist common image processing operations. The rest of the paper is organized as follows. Section II presents the reversible data hiding

methods with an emphasis on obtaining a quality perceived to be high and surveys other schemes capable of resisting manipulations. Section III describes both the CS algorithm with a mean predictor and the robust version of the CS algorithm with feature embedding. Section IV presents the simulation results. Finally, section V provides a brief conclusion.

II. Related Works

This section briefly discusses two kinds of reversible data hiding schemes. The first scheme provides a perceived high quality in marked images with a high embedding rate. The second scheme is robust to image processing operations.

1. Perceptual Quality Schemes

This subsection reviews six outstanding reversible data hiding schemes for high quality images. Kim and others [10] proposed a high-capacity and imperceptible embedding algorithm that exploits the spatial correlation between subsampled images. Based on the predetermined embedding level, this algorithm shifts the histogram and then embeds the data bits by modifying the pixel values. To achieve a desirable perceived quality, Hong and others [11] proposed a reversible data hiding scheme based on modification of prediction errors (MPEs) and proposed a lossless data hiding method. First, they determined pixel values from an input image and then obtained error values. Subsequently, they embedded a secret message into the host image by modifying the prediction errors. The MPE scheme can keep the distortion low when a few messages are embedded. The average PSNR of the marked images generated by the MPE scheme exceeds 48 dB. Sachnev and others [12] suggested a reversible watermarking algorithm based on histogram shifting, prediction, and sorting techniques. By combining rhombus prediction and histogram shifting techniques, a set of sorted prediction errors can be efficiently used to embed bits with less distortion. Lee and others [13] developed an adaptive reversible data hiding approach based on the prediction of difference expansion. Since the difference values between the cover pixels and their corresponding predictive pixels were small, they made use of a large number of smaller difference values to embed data bits. Simulation results demonstrated that this approach can achieve a perceived high quality in marked images. Based on the level 2 IWT, Luo and Yin [14] presented a reversible data hiding scheme. By exploiting the large variance of the IWT coefficient and the utilization of an intelligent histogram shifting technique, the scheme provides a high capacity and imperceptible quality. In addition, the resulting perceived quality is degraded smoothly as the embedding rate increases. Yang and Hu [15] proposed a reversible data hiding scheme using minimum/maximum preserved overflow/underflow avoidance (MMPOUA). The MMPOUA algorithm consists of three main steps: minimum (or maximum) pixel fixing, pixel squeezing, and pixel isolation. Both the pixel squeezing and the pixel isolation supply hiding storage while keeping the amount of distortion low. This algorithm can avoid (or significantly reduce) the overhead bits used to overcome overflow/underflow issues. Simulations showed that the MMPOUA algorithm generates sufficient hiding capacity with a perceived high quality, especially at a moderate rate of embedding.

2. Robustness-Oriented Schemes

For some applications, the marked images generated by reversible data hiding schemes should be robust to manipulations such as image compression, cropping, and noise additions. However, most reversible data hiding schemes are fragile in the sense that the extraction of secret bits can fail upon even a slight alteration to the marked images. Some authors [16], [17] developed robust reversible data hiding techniques to overcome this issue. Ni and others [16] developed a robust lossless data hiding technique based on the patchwork theory, the distribution features of pixel groups, error codes, and the permutation scheme. Although the payload size of this technique cannot exceed 1,024 bits, the resulting images contain no salt-and-pepper noise and the resulting PSNR exceeds 38 dB. Additionally, the marked images generated by the technique are robust to JPEG/JPEG2000 compression. Zeng and others [17] designed a lossless and robust data hiding method by shifting the mathematical difference values of a block. They embedded data bits into blocks by shifting mathematical difference values. Due to the separation of the bit-0-zone and the bit-1-zone and the particularity of the mathematical difference, this method can tolerate non-malicious JPEG compression to some extent. The resulting images, as compared to the images produced by the technique of Ni and others, showed that this method increases hiding capacity at the cost of bit error rate and perceived quality.

III. Proposed Method

In a reversible data hiding scheme for a high-quality image, the proposed CS algorithm with a mean predictor can be performed in a spatial domain. The algorithm consists of two main steps: block-mean removal and pixel-value shifting. To provide a large hiding space, the block-mean removal first generates the difference blocks from an input image. Then, the pixel-value shifting approach further provides hiding storage

while minimizing error. Additionally, a robust reversible data hiding method is generated by conducting the variant of the CS algorithm in the IWT domain. The following subsections provide the details of the proposed CS algorithm.

1. Hiding Data in Spatial Domain

This subsection describes the proposed CS algorithm with a mean predictor, which embeds a secret message into a host medium in a spatial domain. The prediction of a block-mean is first introduced. Then, the procedure of the CS algorithm with a mean predictor that embeds data bits into a host image is described.

A. Block-Mean Prediction

This study employs the CS algorithm with the prediction of block-mean to provide large hiding storage. When a predicted mean is generated, it is subtracted from the pixels in a block to generate a difference block. Thereafter, the secret message can be embedded into these difference blocks. The prediction of the block-mean, that is, the mean predictor m_{MEAN} is defined by

$$m_{\text{MEAN}} = \left\lfloor \frac{m_A + m_B}{2} \right\rfloor, \tag{1}$$

where m_A and m_B are the block-mean of the top block and the left block for the current one, respectively. Notice that the pixel values in the first n-row and n-column of a host image remain intact when the size of a host block is $n \times n$.

B. Data Embedment

Let $C = \{p_{ij}\}_{i=0}^{n\times n-1}$ be the j-th non-overlapping block of size $n \times n$ divided from an input image. A (difference) block can be obtained by $\{\hat{p}_{ij}\}_{i=0}^{n\times n-1} = \{p_{ij}\}_{i=0}^{n\times n-1} - m_j$, where m_j indicates the predicted mean of the j-th block. Then, $\hat{p}_{ij}$ in a difference block shifts to a new value $\tilde{p}_{ij}$ if it satisfies the following criteria:

$$\tilde{p}_{ij} = \begin{cases} \hat{p}_{ij} + \beta, & \text{if } -2\beta \le \hat{p}_{ij} < -\beta, \\ \hat{p}_{ij} - \beta, & \text{if } \beta \le \hat{p}_{ij} < 2\beta. \end{cases} \tag{2}$$

The term β is a control parameter. After CS, data bits are ready to be embedded into $\tilde{p}_{ij} \in \{\hat{p}_{ij}, \tilde{p}_{ij}\}$ with $-\beta < \tilde{p}_{ij} < \beta$, by multiplying $\tilde{p}_{ij}$ by two to obtain $\breve{p}_{ij}$ and adding an input bit to $\breve{p}_{ij}$. Finally, adding m_j to each pixel in the difference block forms a marked block. This procedure is repeated until all of the host blocks have been processed.

C. Data Extraction

First, divide a marked image into a series of non-overlapping blocks that measure $n \times n$. Let $D = \{q_{ij}\}_{i=0}^{n\times n-1}$ be the j-th hidden block of the marked image and m_j be the prediction of the block-mean. The difference pixels of the j-th block are acquired using $\{\hat{q}_{ij}\}_{i=0}^{n\times n-1} = \{q_{ij}\}_{i=0}^{n\times n-1} - m_j$. Data bits can then be extracted from a difference block. If $-2\beta \le \hat{q}_{ij} < 2\beta$, then the data bits can be obtained by applying modulo-2 to $\hat{q}_{ij}$. Subsequently, the pixels $\hat{q}_{ij}$ that hid a data bit can be restored by performing either $\hat{q}_{ij} = \lfloor \hat{q}_{ij} / 2 \rfloor$ if $\hat{q}_{ij} \ge 0$ or $\hat{q}_{ij} = \lceil (\hat{q}_{ij} / 2) - 0.5 \rceil$ if $\hat{q}_{ij} < 0$. The original pixel values can be recovered by adding (or subtracting) β to (or from) $\hat{q}_{ij}$ if $\hat{q}_{ij} \ge 0$ (or $\hat{q}_{ij} < 0$) while the flag of $\hat{q}_{ij}$ is marked. This procedure repeats until all data bits have been extracted. Note that $\lfloor . \rfloor$ and $\lceil . \rceil$ in the descriptions above represent the floor and ceiling functions, respectively.

Figures 1 and 2 present two examples of bit embedding via the CS algorithm with a mean predictor. The parameter β used

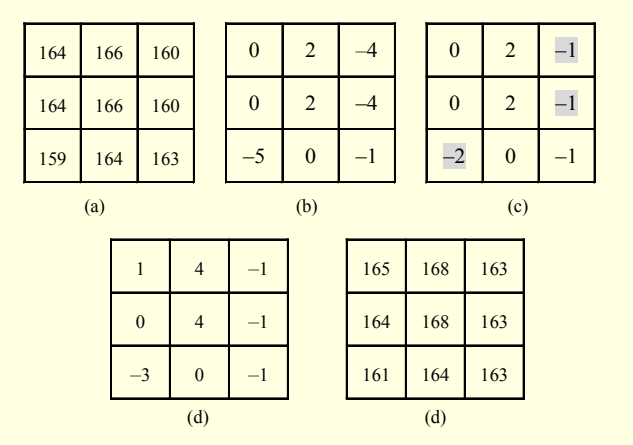


Fig. 1. Example of bit embedding with input bit-stream of 101 001 101: (a) original block, (b) difference block, (c) shifted block, (d) hidden block, and (e) marked block.

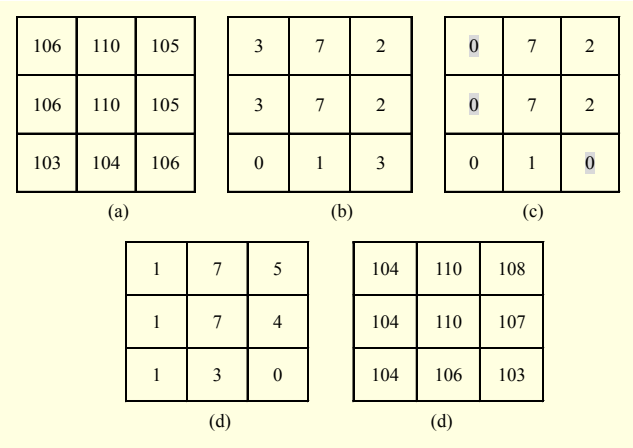


Fig. 2. Example of bit embedding with bit-stream of 111 0110: (a) original block, (b) difference block (with predicted mean of value 103), (c) shifted block, (d) hidden block, and (e) marked block.

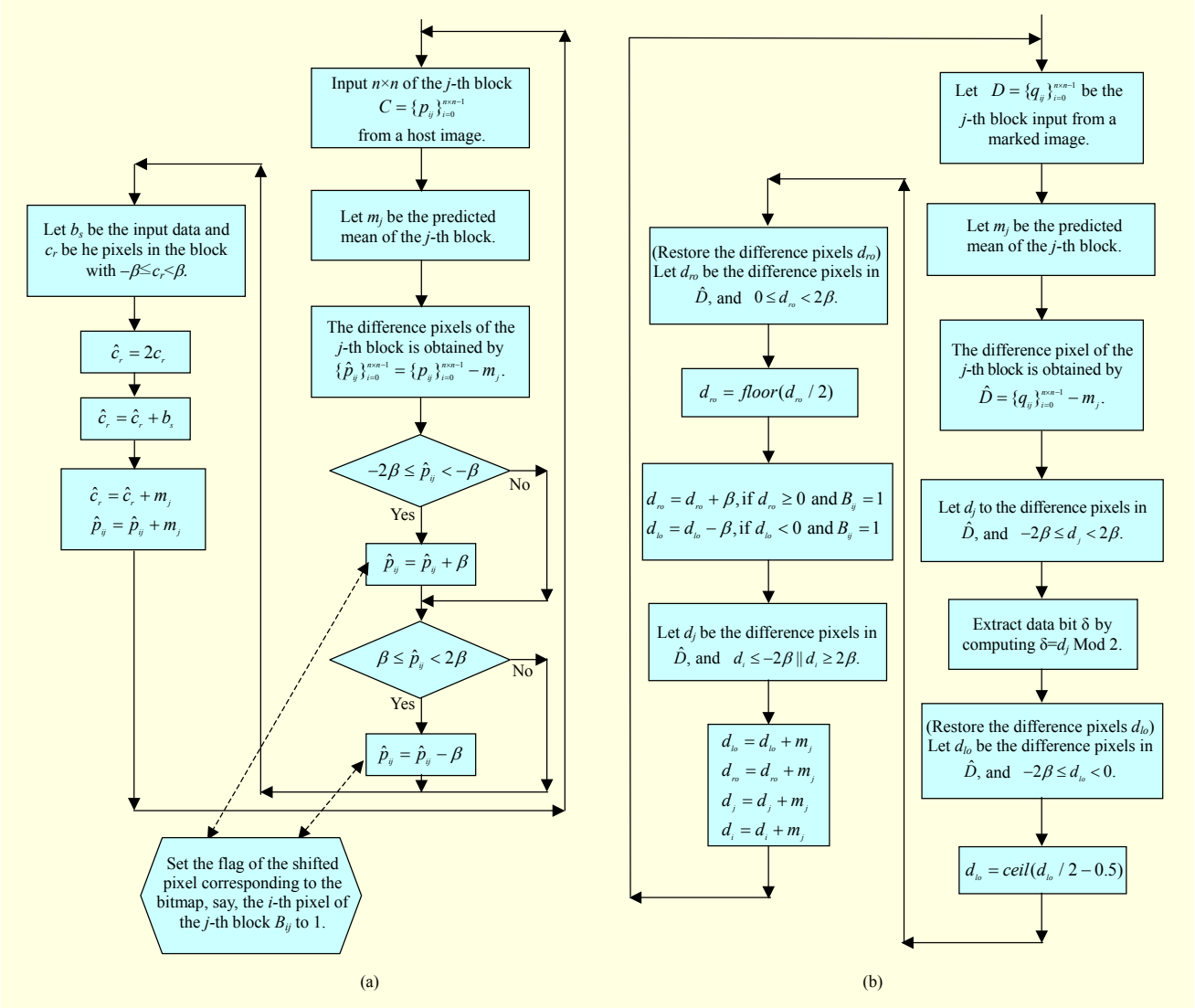


Fig. 3. Flowchart of CS algorithm with mean predictor: (a) encoding and (b) decoding.

here is 3. The predicted mean is 164. Figure 1(a) shows the original (host) block. Figure 1(b) shows a difference block introduced by subtracting each pixel in Fig. 1(a) from the predicted mean. The pixels $\hat{p}_j$ in Fig. 1(b) that satisfy either $\beta\le\hat{p}_j<2\beta$ or $-2\beta\le\hat{p}_j<-\beta$ are shifted by subtracting $\hat{p}_j$ from β or adding $\hat{p}_j$ to β, respectively, as shown by the gray highlighted numbers in Fig. 1(c). Figure 1(d) shows the hidden block. Finally, the marked block in Fig. 1(e) is generated by adding the predicted mean (164) to each value in Fig. 1(d). The mean square error (MSE) computed from Figs. 1(a) and 1(e) is 3.44. Similarly, Fig. 2 illustrates an input with a seven-bit length. In this case, the resluting MSE of the marked block is 3.89.

To recover the original block, a similar reverse process of the CS algorithm with a mean predictor can be performed (Figs. 1(e) and 2(e)). Figure 3 summarizes the encoding and decoding parts of the CS algorithm with a mean predictor.

D. Overhead Information Analysis

A bitmap that indicates whether or not a difference pixel has undergone the shifting process is recorded during bit embedding. The overhead information used in the process of pixel shifting is $\left\lfloor\frac{M}{n}\right\rfloor\times\left\lfloor\frac{N}{n}\right\rfloor\times n^2\le MN$ bits, where the image size is $M\times N$. To help the decoder later extract the data bits, overhead information can be losslessly compressed and sent to the receiver by an out-of-band transmission. Since the overhead information can be independently transmitted to the receiver, it is nearly impossible for third parties (or malicious users) to extract the hidden message and recover the original host image when they steal (or eavesdrop on) the marked

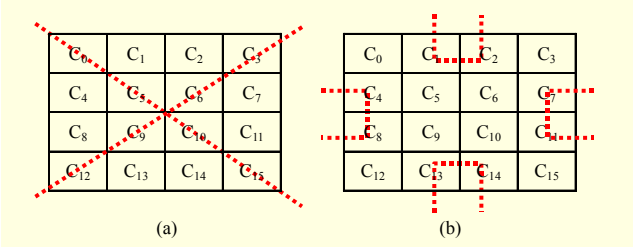


Fig. 4. 4×4 IWT coefficients block: (a) X-sampling coefficients and (b) directional-sampling coefficients.

images. Conversely, if an adversary steals (or eavesdrops on) the marked images, which are generated by existing data hiding methods such as one of the methods in [10]-[14], the hidden message (or watermarks) can possibly be extracted. As a result, the confidentiality of the embedded message can be under severe threat. Even worse, the (original) hidden watermark can be illicitly tampered with and falsified.

To overcome the overflow/underflow issues, a pixel-offset approach can be applied to the spatial domain before embedding. If a pixel p in a host image satisfies either $p < \phi_1$ or $p > \phi_2$ with $\phi_1 < \phi_2$, p can be adjusted to a new value by adding it to or subtracting it from an integer offset δ. Both ϕ_1 and ϕ_2 are predetermined threshold values.

2. Hiding Data in the Transform Domain

To achieve a robust reversible data hiding method, this study embeds a secret message into the transform domain using the variant of the CS algorithm, also known as the robust version of the CS algorithm with feature embedding. Specifically, an input image is first decomposed to the IWT domain. The IWT coefficients can then be acquired using the following two formulas:

$$d_{1,k} = s_{0,2k+1} - s_{0,2k}, \tag{3}$$

$$\boldsymbol{s}_{1,k} = \boldsymbol{s}_{0,2k} + \left\lfloor \frac{\boldsymbol{d}_{1,k}}{2} \right\rfloor, \tag{4}$$

where $s_{j,k}$ and $d_{j,k}$ are the k-th low-frequency and high-frequency wavelet coefficients at the j-th level, respectively [18]. Then, data bits are embedded into the blocks derived from three high subbands: the low-high (LH), high-low (HL), and high-high (HH) subbands of the IWT coefficients. The variant of the CS algorithm consists of two parts, called X-sampling and directional-sampling. A detailed description of this process follows.

A. Bit Embedding

Let $C_j = \{c_{jk}\}_{k=0}^{n^2-1}$ be the j-th block of size $n \times n$ taken from the LH, HL, or HH subband of the IWT domain. Let $C_j = \{\hat{C} \cup \tilde{C}\}$ with $\hat{C} = \{\hat{c}_i \mid i = 0, 3, 5, 6, 9, 10, 12, 15\}$ and $\tilde{C} = \{\tilde{c}_u \mid u = 1, 2, 4, 7, 8, 11, 13, 14\}$ be the X-sampling and directional-sampling coefficients, respectively, as shown in Fig. 4, if n=4. In addition, let

$$C_{jp} = \{\hat{c}_i \mid \beta \le \hat{c}_i < 2\beta\} \tag{5}$$

and

$$C_{jm} = \{\hat{c}_i \mid -2\beta \le \hat{c}_i < -\beta\} \tag{6}$$

be the two focal groups adapted to "carry" data bits. The β used here is a robustness parameter.

The main steps of X-sampling and directional-sampling are the same and are as follows:

Step 1. Input a block C_j not yet processed.

Step 2. If an input bit ϕ=0 and $|C_{jp}| > |C_{jm}|$, then do nothing. This means a bit 0 can be carried by the X- or directional-sampling coefficients without altering their values. Then, proceed to Step 8.

Step 3. If ϕ=0 and $|C_{jp}| = |C_{jm}|$, then add β to the coefficients c_{jk} in C_j with $0 \le c_{jk} < \beta$, mark a flag to the shifted coefficients, and proceed to Step 8.

Step 4. If ϕ=0 and $|C_{jp}| < |C_{jm}|$, then add β to the coefficients in C_{jm}, mark a flag to the shifted coefficients, and proceed to Step 8.

Step 5. If ϕ=1 and $|C_{jp}| < |C_{jm}|$, then do nothing. This means the X- or directional-sampling coefficients carry a bit 1. Then, proceed to Step 8.

Step 6. If ϕ=1 and $|C_{jp}| = |C_{jm}|$, then subtract β from the coefficients c_{jk} in C_j with $-\beta \le c_{jk} < 0$, mark a flag to the shifted coefficients, and proceed to Step 8.

Step 7. If ϕ=1 and $|C_{jp}| > |C_{jm}|$, then subtract β from the coefficients in C_{jp}, and mark a flag to the shifted coefficients.

Step 8. Repeat Step 1 until all blocks of the IWT coefficients have been processed.

Notice that the coefficients that belong to either C_{jp} or C_{jm} have to be changed to $C_{jp} = \{\hat{c}_u \mid \beta \le \hat{c}_u < 2\beta\}$ and $C_{jm} = \{\tilde{c}_u \mid -2\beta \le \tilde{c}_u < -\beta\}$, respectively, when the directional-sampling is employed. The procedures above indicate that each block can carry at most two data bits. Thus, the proposed method can have a total payload of $\lfloor M/2n \rfloor \times \lfloor N/2n \rfloor \times 3 \times 2 \le \frac{3MN}{2n^2}$ bits.

B. Bit Extraction

Let $D_j = \{d_{jk}\}_{k=0}^{n^2-1}$ be the j-th hidden block of size $n \times n$ taken from the LH, HL, or HH subband of the IWT domain derived from a marked image, and $D_j = \{\hat{D} \cup \tilde{D}\}$ with

$\hat{D} = \{\hat{d}_i \mid i = 0, 3, 5, 6, 9, 10, 12, 15\}$ and $\tilde{D} = \{\tilde{d}_u \mid i = 1, 2, 4, 7, 8, 11, 13, 14\}$. Also, let

$$D_{jp} = \{\hat{d}_i(\tilde{d}_u) \mid \beta \le \hat{d}_i(\tilde{d}_u) < 2\beta\} \quad (7)$$

and

$$D_{jm} = \{\hat{d}_i(\tilde{d}_u) \mid -2\beta \le \hat{d}_i(\tilde{d}_u) < -\beta\}. \quad (8)$$

The following steps summarize the bit extraction procedure for both the X-sampling and the directional-sampling.

Step 1. Input a hidden block D_j not yet processed.

Step 2. If $|D_{jp}| > |D_{jm}|$, then a data bit 0 can be extracted. Subtract β from either the coefficients d_{jk} in D_j with $-\beta \le d_{jk} < 0$ or the coefficients in D_{jp} when the corresponding flag of $\hat{d}_i(\tilde{d}_u)$ is set at 1, and proceed to Step 6.

Step 3. If $|D_{jp}| < |D_{jm}|$, then a data bit 1 can be extracted. Add β to either the coefficients d_{jk} in D_j with $0 \le d_{jk} < \beta$ or the coefficients in D_{jm} when the corresponding flag of $\hat{d}_i(\tilde{d}_u)$ is set at 1, and proceed to Step 6.

Step 4. If $|D_{jp}| = |D_{jm}|$ and the flag of the coefficients d_{jk} in D_j with $-\beta \le d_{jk} < 0$ is set at 1, a bit 0 can be extracted. Proceed to Step 6.

Step 5. If $|D_{jp}| = |D_{jm}|$ and the flag of the coefficients d_{jk} in D_j with $0 \le d_{jk} < \beta$ is set at 1, a bit 1 can be extracted.

Step 6. Repeat Step 1 until all hidden bits have been extracted.

IV. Experiment Results

The experiments in this study use several grayscale images measuring 512×512 as host images (Fig. 5). A quarter of the host image Lena is used as the test data. The following subsections examine simulations generated by the CS algorithm with a mean predictor and a variant of the CS algorithm. The control parameter β is not a fixed value.

1. CS Algorithm with a Mean Predictor

To demonstrate the hiding performance of the CS algorithm with a mean predictor, Fig. 6 depicts the relationship between the payload and PSNR for several images. Figure 6 indicates that an average PSNR value of 57.49 dB is achieved with the payload of 0.24 bpp on all images except Baboon. However, an optimal PSNR value of 60.96 dB can be achieved at 0.11 bpp on Baboon. In addition, the average payload of the test images is 0.84 bpp with a PSNR value of 34.99 dB when β is set at 10. The size of the block is 3×3. The PSNR is defined by

$$PSNR = 10 \times \log_{10} \frac{255^2}{MSE}, \quad (9)$$

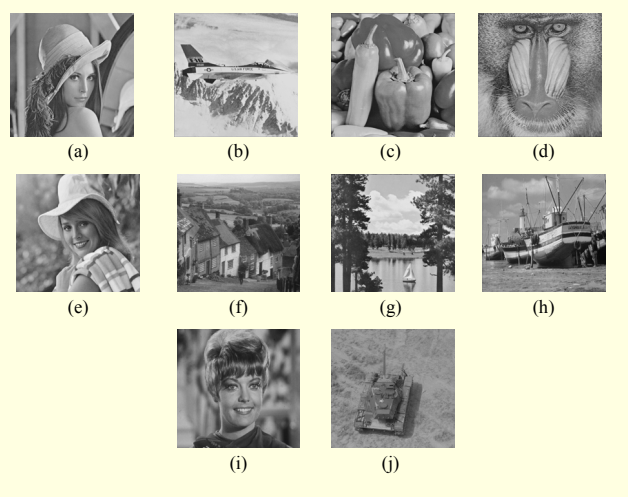


Fig. 5. Host images: (a) Lena, (b) Jet, (c) Peppers, (d) Baboon, (e) Elaine, (f) Goldhill, (g) Scene, (h) Boat, (i) Zelda, and (j) Tank.

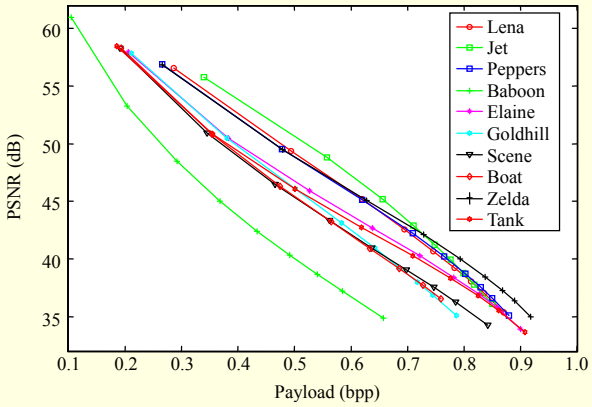


Fig. 6. Trade-off between PSNR and payload for proposed method.

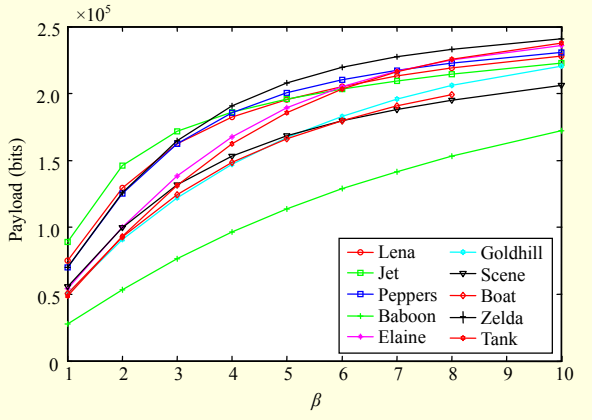


Fig. 7. Relationship between payload and β for proposed method.

where $MSE = \frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M}(\hat{x}(i,j) - x(i,j))^2$. Here, $x(i,j)$

and $\hat{x}(i, j)$ denote the pixel values of the original image and the marked image, respectively. Moreover, Fig. 7 reveals the relationship between the payload and β, and it indicates that the payload is smoothly increased as β is enlarged.

This study compares the proposed method with several brilliant schemes [10]-[15]. Table 1 provides a performance comparison of these methods. The proposed method provides the largest payload among these methods and a better PSNR than all six schemes. Table 1 shows that the hiding capacity provided by the proposed method is nearly six times that achieved by Kim and others' approach [10], approximately two times that achieved by Hong and others' technique [11], and three times larger than that achieved by Lee and others' scheme [13]. Furthermore, in the case of a small hiding capacity, Table 2 indicates that the proposed method achieves a better PSNR and payload than Hong and others' technique [11] and Sachnev and others' algorithm [12], as well as Luo and Yin's scheme [14]. Since the optimal PSNRs for the methods of Kim and others [10] and Lee and others [13] do not exceed 50 dB, neither method is included in Table 2. Although the PSNR for the proposed method is slightly less than that for Yang and Hu's technique [15], the payload provided by our method is approximately eight times that achieved by Yang and Hu's technique [15].

Table 1. Performance comparison of various methods with PSNR around 48 dB.

Alg.	Bit rate/ PSNR				
	Lena	Jet	Baboon	Boat	Average
[10]	0.07/48.9	0.12/49	0.02/48.7	0.08/48.9	0.07/48.88
[11]	0.33/48.93	0.27/48.79	0.06/48.29	0.17/48.53	0.21/48.64
[12]	0.27/47.74	0.42/48.17	0.08/49.45	–	0.26/48.45
[13]	0.14/48.54	0.19/48.54	0.14/48.54	0.06/48.54	0.13/48.54
[14]	0.49/48.2	0.29/48.5	0.29/48	0.38/47.6	0.36/48.08
[15]	0.20/48.48	0.30/48.26	0.05/48.62	0.11/48.57	0.17/48.48
Proposed	0.49/49.38	0.57/48.83	0.20/53.26	0.38/50.50	0.41/50.49

Table 2. Performance comparison of various methods with small hiding capacity.

Alg.	Bit rate/ PSNR				
	Lena	Jet	Baboon	Boat	Average
[11]	0.03/61.80	0.07/55.14	0.02/51.79	0.03/56.79	0.04/56.38
[12]	0.08/55.29	0.15/55.30	0.04/54.23	–	0.09/54.94
[14]	0.05/54.25	0.05/51.35	0.05/50	0.05/51.65	0.05/51.81
[15]	0.03/58.49	0.05/56.39	0.01/64.10	0.02/60.95	0.03/58.98
Proposed	0.29/56.57	0.34/55.81	0.10/60.96	0.19/58.28	0.23/57.91

2. Variant of CS Algorithm

This subsection demonstrates the experiment results generated by the robust version of the CS algorithm with feature embedding. Half of the images in Fig. 5 are used as the host images. The size of the block is 4×4. Figure 8 shows the trade-off between PSNR and robustness parameter β. The larger the value of β, the lower the value of the PSNR. Note that the larger the β, the better the robustness for the proposed method.

This study also compares two outstanding techniques, namely, the Ni and others' algorithm [16] and the Zeng and others' scheme [17], with the proposed method. Table 3 shows the performance comparison of these methods on three test images. The proposed method clearly provides the largest payload among these methods and achieves a better PSNR than the other two techniques.

To demonstrate the robustness performance of the proposed method, Table 4 provides examples of extracted watermarks (size of 55×55 with 8 bits/pixel, 2 colors) after various manipulations. The bit correct ratio (BCR) is also included.

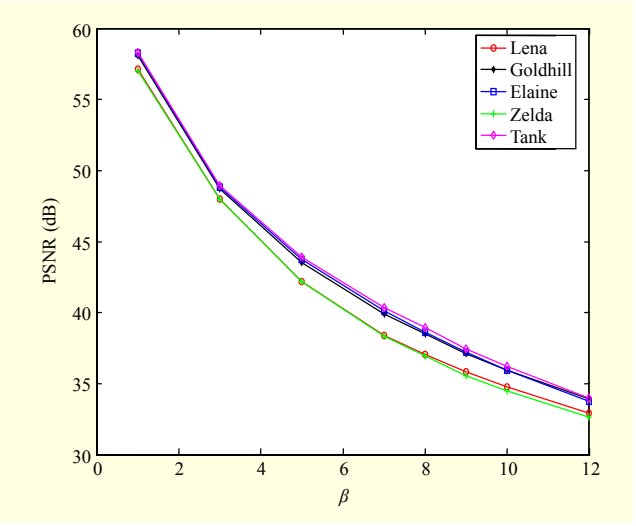
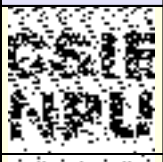
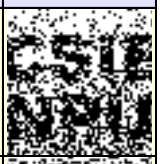
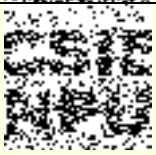
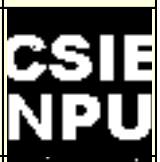
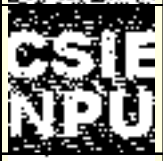


Fig. 8. Relationship between PSNR and β.

Table 3. Payload/ PSNR comparison of various methods.

Image	Bit rate/ PSNR		
	Ni and others [16]	Zeng and others [17]	Proposed (with β =5)
Lena	6,336/40.19	16,384/38.07	24,576/42.18
Zelda	4,480/40.47	16,384/38.09	24,576/42.20
Goldhill	6,336/40.18	16,384/38.10	24,576/43.50
Average	5,717/40.28	16,384/38.09	24,576/42.63

Table 4. Examples of watermarks extracted from image Lena (β=12).

Attack	Survived watermark	Attack	Survived watermark	Attack	Survived watermark	Attack	Survived watermark
Null attack BCR = 100%		Gaussian noise (4%) BCR = 74.22%		Brightness (100%) BCR = 82.31%		Winding BCR = 75.97%	
Cropping (50%) BCR =80.63 %		Equalized BCR = 78.28%		Brightness (–100%) BCR = 90.94%		Zigzag BCR = 70.15%	
JPEG2000 (CR*=10) BCR=61.55%		Mean filtering (3×3) BCR = 99.77%		Contrast (45%) BCR = 80.86%		Inversion BCR = 0.06%	
JPEG (CR=3.88) BCR=77.39%		Median filtering (3×3) BCR = 99.77%		Contrast (–90%) BCR = 9.98%		Poster edges BCR = 27.74%	
Uniform noise (5%) BCR = 78.65%		Quantization† BCR = 98.05%		Posterized (16 level) BCR = 94.45%		Interleaving BCR = 98.05%	

*CR stands for compression ratio, which is defined as the ratio of the size of a host image to that of a compressed image.
†The last three bits of the pixel in a marked image were truncated.

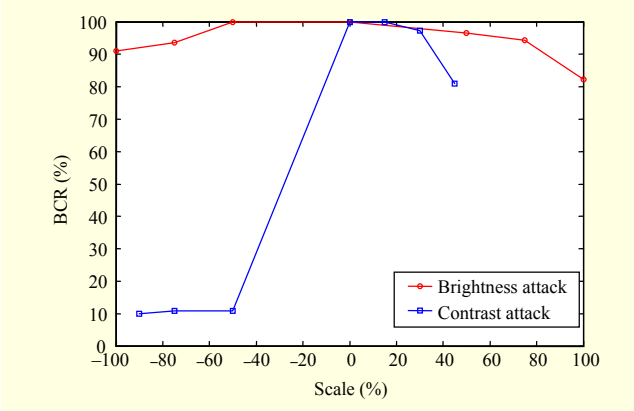


Fig. 9. BCR performance of proposed method under brightness/contrast (with various scales) attacks.

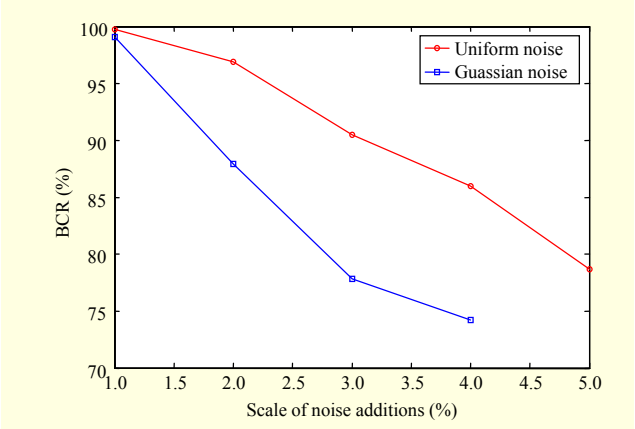


Fig. 10. BCR performance of proposed method under uniform/Gaussian noise additions (with various scales) attacks.

The BCR is defined by

$$BCR = \left(\sum_{i=0}^{ab-1} \overline{w_i \oplus \tilde{w}_i} \Big/ a \times b \right) \times 100\%, \qquad (10)$$

where w_i and $\tilde{w}_i$ represent the values of the original watermark and the extracted watermark, respectively, and the size of a watermark is $a \times b$. The BCR for an extracted watermark is 100% if a marked image remains intact (null attack). Table 4 shows that most of the extracted watermarks are easily recognized. Although the BCR for watermarks that suffer from attacks, such as cropping, JPEG2000 processing, sharpening, poster edges, interleaving, and inversion, is not

high, the images are identifiable. Though the BCR for the watermark extracted from an image attacked by inversion is only 0.06%, the image is recognizable. Furthermore, Fig. 9 depicts the BCR for the watermarks extracted from the marked images whose brightness and contrast were manipulated. The BCR performance of watermarks suffering from a brightness attack was better than that of watermarks suffering from a contrast attack. In other words, the marked images generated by the proposed method are more robust to brightness attacks than contrast attacks. Similarly, Fig. 10 indicates that the proposed method is more robust to attack from uniform noise than Gaussian noise additions.

V. Conclusion

This paper presented two reversible data hiding schemes based on the coefficient shifting (CS) algorithm. First, high-performance reversible data hiding using the CS algorithm with a mean predictor in the spatial domain was proposed to provide a high payload while minimizing distortion. Second, the variant of the CS algorithm based on the IWT domain was presented to guard against manipulations. Simulation results demonstrated that both the payload and the PSNR generated by the CS algorithm with the use of prediction were superior to those generated by existing techniques. In addition, the marked images generated by the variant of the CS algorithm were tolerant of various attacks such as JPEG2000 compression, JPEG compression, noise additions, (edge) sharpening, and so on. Our future study will focus on the reduction of overhead bits while maintaining sufficient perceived quality with a high payload.

Acknowledgments

The authors would like to thank the editors and anonymous reviewers for providing valuable comments that helped to improve the content of the paper.

References

[1] F.Y. Shih, *Digital Watermarking and Steganography: Fundamentals and Techniques*, Boca Raton, FL: CRC Press, 2008.

[2] I.J. Cox et al., *Digital Watermarking and Steganography*, 2nd ed., Burlington, MA: Morgan Kaufmann, 2008.

[3] Z.G. Qu et al., "Novel Quantum Steganography with Large Payload," *Optics Commun.*, vol. 283, no. 13, 2010, pp. 4782-4786.

[4] S. Wang, B. Yang, and X. Niu, "A Secure Steganography Method Based on Genetic Algorithm," *J. Inf. Hiding Multimedia Signal Process.*, vo1. 1, no. 1, 2010, pp. 28-35.

[5] K. Yamamoto and M. Iwakiri, "Real-Time Audio Watermarking Based on Characteristics of PCM in Digital Instrument," *J. Inf. Hiding Multimedia Signal Process.*, vol. 1, no. 2, 2010, pp. 59-71.

[6] C.Y. Yang et al., "A Simple Digital Watermarking by the Adaptive Bit-Labeling Scheme," *Int. J. Innovative Computing, Inf. Control*, vol. 6, no. 3, 2010, pp. 1401-1410.

[7] J. Tian, "Reversible Data Embedding Using a Difference Expansion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, 2003, pp. 890-896.

[8] A.M. Alattar, "Reversible Watermark Using the Difference Expansion of a Generalized Integer Transform," *IEEE Trans. Image Process.*, vol. 13, no. 8, 2004, pp. 1147-1156.

[9] D.M. Thodi and J.J. Rodriguez, "Expansion Embedding Techniques for Reversible Watermarking," *IEEE Trans. Image Process.*, vol. 16, no. 3, 2007, pp. 721-730.

[10] K.S. Kim et al., "Reversible Data Hiding Exploiting Spatial Correlation Between Sub-sampled Images," *Pattern Recognition*, vol. 42, 2009, pp. 3083-3096.

[11] W. Hong, T.S. Chen, and C.W. Shiu, "Reversible Data Hiding for High Quality Images Using Modification of Prediction Error," *J. Syst. Software*, vol. 82, 2009, pp. 1833-1842.

[12] V. Sachnev et al., "Reversible Watermark Algorithm Using Sorting and Prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 7, 2009, pp. 989-999.

[13] C.F. Lee, H.L. Chen, and H.K. Tso, "Embedding Capacity Raising in Reversible Data Hiding Based on Prediction of Different Expansion," *J. Syst. Software*, vol. 83, 2010, pp. 1864-1872.

[14] X.R. Luo and T.L.Yin, "Reversible Data Hiding Exploiting Variance in Wavelet Coefficients," *J. Computer Sci. Technol.*, vol. 11, no. 1, 2011, pp. 27-33.

[15] C.Y. Yang and W.C. Hu, "High-Performance Reversible Data Hiding with Overflow/Underflow Avoidance," *ETRI J.*, vol. 33, no. 4, Aug. 2011, pp. 580-588.

[16] Z. Ni et al., "Robust Lossless Image Data Hiding Designed for Semi-Fragile Image Authentication," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 4, 2008, pp. 497-509.

[17] X.T. Zeng, L.D. Ping, and X.Z. Pan, "A Lossless Robust Data Hiding Scheme," *Pattern Recognition*, vol. 43, 2010, pp. 1656-1667.

[18] A.R. Calderbank et al., "Wavelet Transforms that Map Integers to Integers," *Appl. Computational Harmonics Anal.*, vol. 5, no. 3, 1998, pp. 332-369.

Ching-Yu Yang received his BS in electronic engineering from the National Taiwan Institute of Technology, Taiwan, in 1983 and his MS in electrical engineering from National Cheng Kung University, Taiwan, in 1990. In 1999, he received his PhD in computer and information science from National Chiao Tung University, Hsinchu, Taiwan. From 1999 to 2005, he was a senior engineer at Chunghwa Telecom. Co. Ltd., Taiwan. He joined the Computer Science and Information Engineering Department at National Penghu University of Technology in February 2005 and is currently an associate professor there. His research interests include image processing, pattern recognition, and data hiding.

Chih-Hung Lin received his BS and MS from the Department of Information Engineering and Computer Science at Feng Chia University, Taichung, Taiwan, in 1995 and 1998, respectively, and his PhD from the Department of Computer Science and Engineering at National Sun Yat-Sen University, Kaohsiung, Taiwan, in 2006. He has been an assistant professor at the Graduate Institute of Mathematics and Science Education, National Chiayi University, Chiayi, Taiwan, since 2011. He obtained SCJP (Sun Certified Java Programmer) and MCTS (Microsoft Certified Technology Specialist) certification in 2007 and 2009, respectively. His research interests include digital watermarking, data hiding, multimedia application, image processing, digital contents design in education, and e-learning.