Journal of Information Hiding and Multimedia Signal Processing ©2010 ISSN 2073-4212
Ubiquitous International Volume **1**, Number **2**, April **2010**

Reversible Data Hiding by Coefficient-bias Algorithm

Ching-Yu Yang[a],Wu-Chih Hu[a] and Chih-Hung Lin[b]

[a]Dept. of Computer Science and Information Engineering, National Penghu University
No. 300, Liu-Ho Rd., Magong, Penghu, 880 Taiwan
{ chingyu; wchu}@ npu.edu.tw

[b]Dept. of Computer Science and Information Engineering
Southern Taiwan University
No.1, Nantai St., Yongkang City, Tainan County, 710 Taiwan
chuck@mail.stut.edu.tw

Received September 2009; revised January 2010

Abstract. *A simple lossless data hiding method based on the coefficient-bias algorithm by embedding bits in both spatial domain and frequency domain is proposed. In spatial domain, each pixel in a host image is first subtracted from the block-mean. Then, a stego-image is generated by embedding a large amount of bits (or the primary message) in the mean-removed blocks via the coefficient-bias algorithm. To provide an extra security and robustness, the stego-image is transformed to frequency domain by integer wavelet transform (IWT). A secondary watermark is hidden in the low-high (LH) and high-low (HL) subbands of IWT domain by the proposed algorithm. Simulations show that both the perceptual quality and hiding capacity are not bad. Moreover, the resultant images introduced by the proposed method are tolerant of the attacks such as JPEG2000, JPEG, brightness, and inverting.*
Keywords: reversible data hiding, coefficient-bias algorithm

1. **Introduction.** With the proliferation of computer networks and architectures, and ubiquitous broadband services provided by the Internet Service Providers (ISPs), people are capable of surfing on the Internet at an acceptable cost. It is also convenient to people (or parties) to share the resources and conduct commercial activities on the Internet. However, the hackers might exploit the servers (or a client system) to dig out a piece of valuable data, such as credit card number, bank account and password, which are supposed to not be exposed in public. Moreover, the adversaries could perform man-in-the-middle attacks to eavesdrop, tamper or falsify data which transmitted between two parties. Therefore, how to protect (or secure) a significant (or privacy) data from being stolen or illegal alteration becomes an important issue. Most organizations utilize encryption/decryption techniques to protect data. But, the exposure of a private key may result in insecurity of the confidential data. On the other hand, data hiding provides an alterative solution to guard against illegal behavior from the adversaries. Generally speaking, data hiding could be classified into fragile watermarking and robust watermarking [1-3]. The fragile watermarking approaches [4-6] have the capability of hiding a number of bits in a host medium while the resultant perceived quality is good. But, the approaches are vulnerable to common image processing operations. To resist manipulations, some robust watermarking schemes [7-9] are presented in the literature. However, most of the schemes provide a limited size of the payload. Due to a host medium is a valuable object such as low enforcement, medical imaging system, or geographic information system, it is not

allowed a host medium to be damaged after digital watermarking. Several researchers [10-19] suggested lossless data hiding techniques, also referred to as reversible watermarking techniques, to provide solutions for completely restored the original host medium. Using the idea of the difference expansion (DE), Tian [10] explored the redundancy in the host images and developed a high-capacity and low-distortion lossless data hiding technique. Alattar [11] used DE of vectors, namely, the generalized DE of an arbitrary size, instead of pairs and presented a reversible hiding scheme. Ni et al. [12] utilized the zero (or the minimum) points of the histogram to embed data bits in a host medium. Although the resulting peak signal-to-noise ratio (PSNR) is about 48.20 dB, the hiding capacity is not good enough. Lin et al. [14] took advantage of the block difference histogram of a host medium and developed a reversible watermarking scheme. Experiments reveal that a high hiding capacity and visual quality is achieved by their method. Lin and Hsueh [16] used the three-pixel block differences technique to gain a lossless data hiding method. In the host image, an absolute difference between a pair of pixels in selected to embed a message if the number of pixel pairs with the difference in the image is the largest. In the best case, two data bits can be embedded in a three-pixel block. Their simulations show that the average embedding rate can be up to 2.08 bits per pixel (bpp). Tseng and Chang [17] proposed a reversible watermarking algorithm using the technique of the shiftable pixel pairs. The extended difference expansion algorithm does supply a great number of hiding capacity without making noticeable distortion. Hsiao et al. [18] proposed an elegant block-based reversible data hiding method. An input image is first divided into two categories: data embedding block and overhead information embedding block. The embedding blocks are used hide a message while the location map and other auxiliary information are embedded into overhead embedding blocks. Simulations indicate that the bit rate of their method is about 1.00 bpp with an acceptable perceptual quality on some smooth images. Tsai et al. [19] utilized predictive coding and histogram shifting to further improve the performance of the method in Ref. [10]. In addition, the technique has good performance in hiding capability and resulting perceived quality when the resultant images are introduced from the medical images. From the above review we can see that most of the schemes conducted in spatial domain achieve a high payload size. As described previously, the schemes embed bits in spatial domain are vulnerable to manipulations. Namely, the hidden data is incapable of being extracted if even a slight alteration was imposed to the marked images. To provide a larger hiding capacity with a better robust performance, we embed data bits in both spatial domain and frequency domain.

The rest of the paper is organized as follows. The coefficient-bias algorithm, which includes the kernel parts of the algorithm, namely, data encoder and data decoder, as well as the process of hiding data in spatial domain and in frequency domain are specified in Section 2. Section 3 presents the simulations. Section 4 gives a brief conclusion.

2. **Proposed Coefficient-bias Algorithm.** The idea of the coefficient-bias algorithm is to embed data bits in both spatial domain and frequency domain. That is, a stego-image is first generated by embedding the primary message in the spatial domain. Then, the stego-image is decomposed to IWT domain for hiding the secondary watermark. The schematic view of the proposed method is depicted in Fig. 1. Some notations shown in the figure are defined in the following:

S_H : *A host image.*

H_S : *A stego − image contains a secret message.*

F_S : *The IWT domain obtained from* H_S.

S_M : *A mixed image contains a secret message and a watermark.*

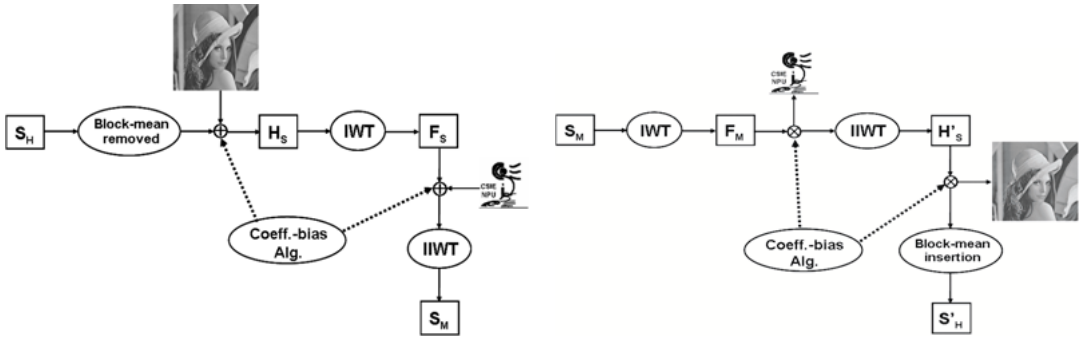
F_M : *The IWT domain obtained from* S_M.
H'_S : *An intermediate image after the extraction of a watermark.*
S'_M : *A restored image after the extraction of a secret message.*

Note that H'_S and S'_H are equivalent to H_S and S_H, respectively, if the mixed images are intact without manipulations by the third parties. Consequently, both the watermark and the secret message would be losslessly extracted and the host images are perfectly restored at receiver site. The details of the coefficient-bias algorithm are specified in the following sections.

2.1. **Data embedding.** Without loss of generality, let $C = \{c_j\}_{j=0}^{(n\times n)-1}$ be an input block of size $n \times n$ and δ be the input data bits. If there exists a coefficient $c_l \in C$ which satisfying $c_l \leq -\beta$, then subtract c_l from $(2^k-1)\beta$. If there also exists a coefficient $c_r \in C$ which satisfying $\beta \leq c_r$, then add c_r to $(2^k-1)\beta$. The β is a control parameter and k is a positive integer. In other words, a new coefficient $\widehat{c}$ in the (host) block C can be obtained by the following rules:

$$\widehat{c} = \begin{cases} c_l - (2^k-1)\beta, & if\ c_l \leq -\beta \\ c_r - (2^k-1)\beta, & if\ c_l \geq -\beta. \end{cases} \tag{1}$$

Generally speaking, the larger the value of β, the better the hiding capability, however, the lower the PSNR. After coefficients adjustment, data bits are ready to be embedded in the block. Multiply the coefficients $c_{dr} \in C$ which satisfying $0 \leq c_{dr} < \beta$ by 2^k to obtain $\widehat{c}_{dr}$. Add δ to $\widehat{c}_{dr}$. In addition, multiply the coefficients $c_{dl} \in C$ which satisfying $-\beta < c_{dl} < 0$ by 2^k to obtain $\widehat{c}_{dl}$. Subtract $\widehat{c}_{dl}$ from δ. The procedure is repeated, until all the blocks have been processed.



(a) Transmitter (b) Receiver

FIGURE 1. The block diagram of the proposed method. (a) Transmitter (b) Receiver

2.2. **Data extraction.** At the receiver, input a (hidden) block D of size $n \times n$ not processing yet. If there exists a coefficient $d_j \in D$, which satisfies $-2^k\beta < d_j < 2^k\beta$, divide d_j by 2^k. The hidden bits can be obtained from the residual while the coefficients which originally located between $-\beta$ and β can be restored from the quotinent. Moreover, restore the coefficients which originally less than or equal to $-\beta$ by adding d_l, which satisfies $d_l \leq -2^k\beta$ to $(2^k-1)\beta$; restore the coefficients which originally greater than or equal to β by subtracting d_r, which satisfies $d_r \geq 2^k\beta$ from $(2^k-1)\beta$. The procedure is repeated until all data bits are extracted. The process of data embedding and extraction is summarized in Fig. 2. (For a clear specification of the proposed algorithm, as shown in Fig. 2, the value of k is set at 1.)

2.3. **Hiding data in spatial domain.** Hiding a large amount of bits in spatial domain is the major aim of the phase I of the proposed method. To achieve the goal, an input image is divided into an $m \times m$ non-overlapping blocks. The pixels in each block are then subtracted from the mean of the block. Let M_k the average value of the pixels in the kth block B_k of an image. The M_k of Bk is computed and recorded. Each pixel p_j in B_k is subtracted from M_k. After subtraction, the kth mean-removed block $\widehat{B}_k = \{\widehat{p}_j\}_{j=0}^{(m\times m)-1}$ with $\widehat{p}_j = p_j - M_k$ is ready to be used for hiding bits. Subsequently, the process of encoder, as shown in Fig. 2(a), is employed to the mean-removed block $\widehat{B}_k$. It goes without saying that the decoder shown in Fig. 2(b) is performed at the receiver during bits extraction. Another control parameter γ used here acts the same role as the β mentioned at Sec. 2.1 does.

Figures 3-4 present examples of bit embedding (in the spatial domain) using the proposed method. These figures illustrate the case of 7-bit and null-bit hidden, respectively. The control parameter k used here is 1 and γ is set at 3. In Fig. 3(a), we assume that the divided block measures 3×3. Figure 3(b) shows a difference block introduced by subtracting each pixel in Fig. 3(a) from the mean (of value 209). To keep distortion low, the difference pixels $\widehat{p}_j$ which satisfy either $\widehat{p}_j \leq -\gamma$ or $\widehat{p}_j \geq \gamma$ are isolated by subtracting $\widehat{p}_j$ from γ or adding $\widehat{p}_j$ to γ, respectively, as shown by the gray highlighted numbers in Fig. 3(c). Figure 3(d) shows the bit-hidden block. Finally, the marked block in Fig. 3(e) was generated by adding the mean (209) to each value in Fig. 3(d). Note that the mean square error (MSE) computed from Fig. 3(a) and 3(e) was 5.56. It is obvious that the hidden bits can be easily extracted in the reverse process at the receiver. Besides, none of data bits is hidden in the block, as shown in Fig. 4(a), because all pixels were isolated, as shown in Fig. 4(b)-(c). However, the MSE evaluated from Fig. 4(a) and 4(d) is 9.

The trade-off between PSNR and bit rate for the proposed method using various γ in the spatial domain is drawn in Fig. 5. Figure 5 indicates that the (average) maximum of PSNR is about 48.40 dB under a bit rate of 0.096 bpp. On the other hand, the (average) maximum bit rate is achieved at 0.907 bpp with the PSNR of value 30.02 dB. Notice that to help the receiver later to extract bits successfully, the overhead information, namely, the mean of each block can be either embeded in a host medium or by means of out-of-band transmitted to the receiver.

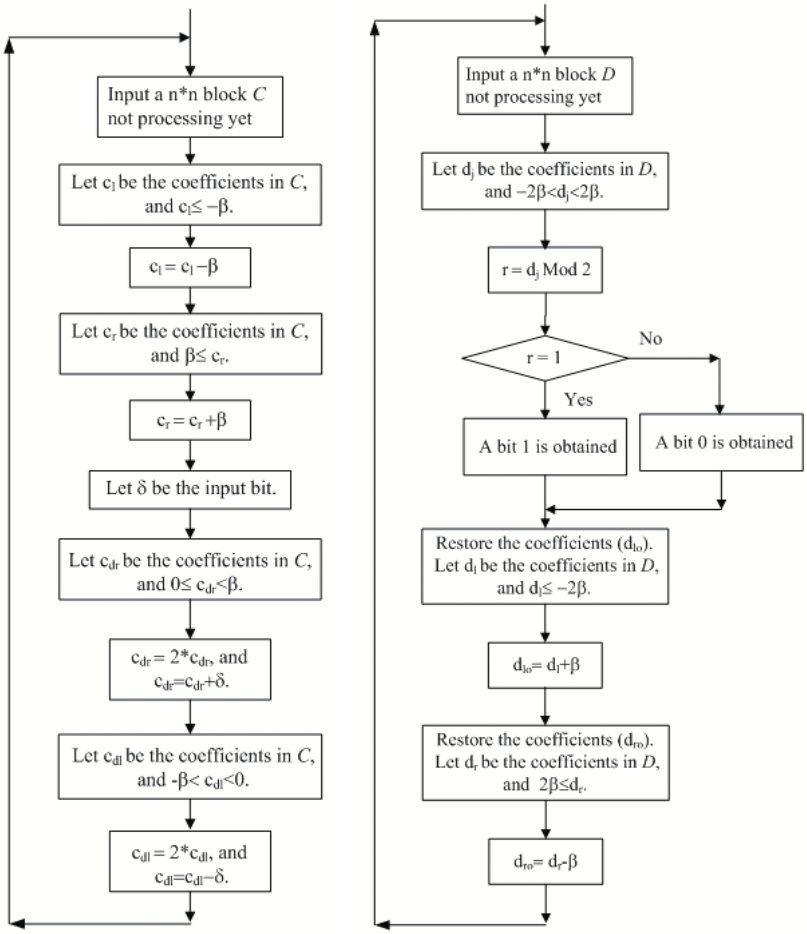
2.4. **Hiding data in frequency domain.** To provide the capability of resisting image processing operations, the phase II of the proposed method is used to embed a watermark in the IWT domain. That is, a stego-image introduced at phase I is decomposed to frequency domain by IWT. The IWT coefficients are acquired by the following two formulas:

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

and

$$s_{1,k} = s_{0,2k} + \lfloor \frac{d_{1,k}}{2} \rfloor \tag{3}$$

The $\lfloor x \rfloor$ is a floor function. The bits hiding procedure, as shown in Fig. 2(a), is applied to embed a watermark in the low-high (LH) and high-low (HL) subbands of IWT domain. At the receiver, a watermark can be extracted from the LH and HL subbands of IWT domain via the procedure shown in Fig. 2(b). Note that the process of phase II is much simpler than that of phase I owing to the former requires no overhead information during data embedding and extraction.



(a) Encoder (b) Decoder

FIGURE 2. The encoder part and decoder part of the coefficient-bias algorithm. (a) Encoder and (b) Decoder.

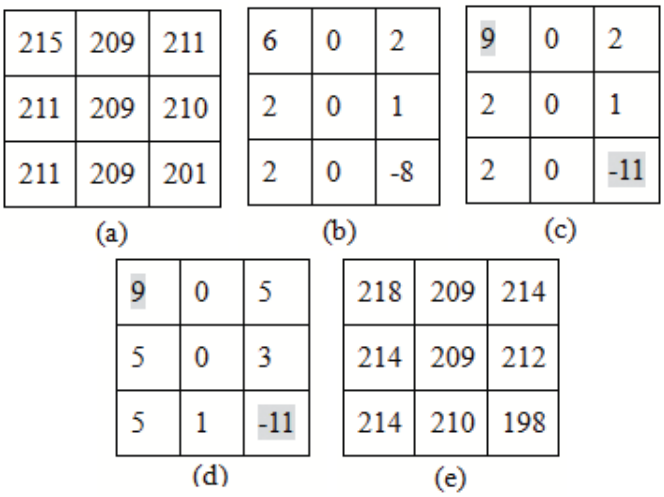


FIGURE 3. Example of 7-bit embedding with a bit-stream of 0110111. (a) 3×3 block of the original block, (b) difference block, (c) isolated block, (d) bit-hidden block, and (e) marked block.

176	77	80
169	162	87
101	167	161

(a)

45	-54	-51
38	31	-44
-30	36	30

(b)

48	-57	-54
41	34	-47
-33	39	33

(c)

179	74	77
172	165	84
98	170	164

(d)

FIGURE 4. Example of a (texture) block contains no dtat bits (with mean=131). (a) 3×3 block of the original block, (b) difference block, (c) isolated block (or null-bit-hidden block), and (d) resultant block.

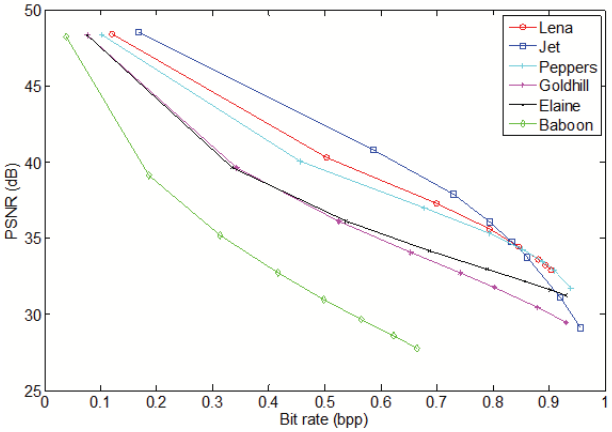
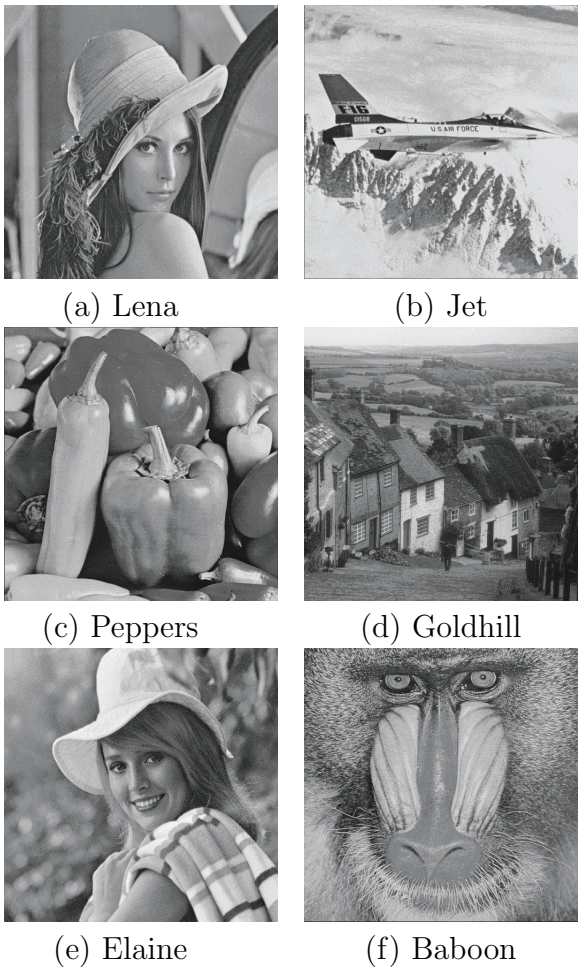


FIGURE 5. The PSNR and bit rate performance generated by the proposed method in the spatial domain.

Notice as well the extraction of primary message would be failed as the mixed images undergone manipulations, however, it is possible for the extracted watermark to survive from the attacks and recognized at the receiver. In other words, the benefits for the proposed method hiding bits in both spatial and transform domains are the providing of a larger hidden capacity and robustness.

3. **Experimental Results.** Several 512×512 gray-scale images were used as the host images. An image Lena sized 256×256 was used as a test data. In addition, a logo-image of measuring 117×117 with 8-bit per pixel was used as the test watermark. An integer k used here is 1. The mixed images generated by embedding test data in the host images via the proposed method are shown in Fig. 6. The block size used in spatial domain and frequency domain are 3×3 and 4×4, respectively. It can be seen from Fig. 6 that the perceived quality is not bad. Two control parameters γ and β are all set at 3. Moreover, the relationship between PSNR and bit rate for the proposed method is depicted in Fig. 7. From the figure we can see that the average PSNR for a half mixed images is about 30.35 dB with a bit rate of 1.20 bpp.

To reveal how the performance of our method affected by the above two control parameters, the trade-off between PSNR and bit rate for the proposed method coupled with various combinations of γ and β is given in Fig. 8. It is clear that the PSNR is approximately linear declined as the size of the payload increased. Namely, the larger the value of γ is used, the larger the hiding capacity is obtained, but with a smaller PSNR values.



(a) Lena (b) Jet

(c) Peppers (d) Goldhill

(e) Elaine (f) Baboon

FIGURE 6. The mixed images generated by the proposed method. (a) Lena, (b) Jet, (c) Peppers, (d) Goldhill, (e) Elaine, and (f) Baboon.

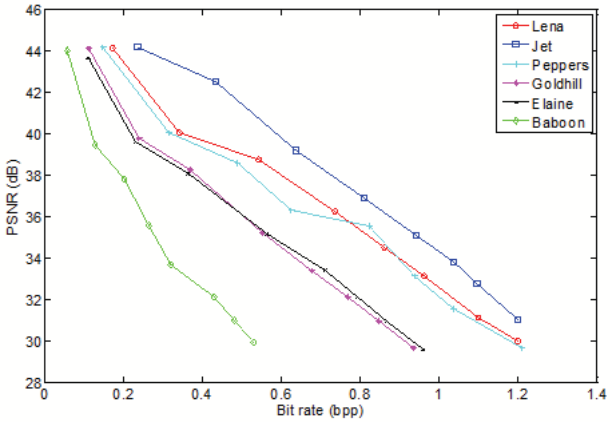


FIGURE 7. Trade-off between PSNR and bit rate for the proposed method.

For comparison, the reported lossless data hiding schemes, namely, Lin et al.'s algorithm [14], Lin and Hsueh's technique [16], and Hsiao et al.'s approach [18] are compared with our method. Their hiding performance is listed in Table 1. From Table 1 we can see that the average PSNR and bit rate generated by the proposed method are slightly less than those genertaed by Lin and Hsueh's technique [16], but are superior to the other

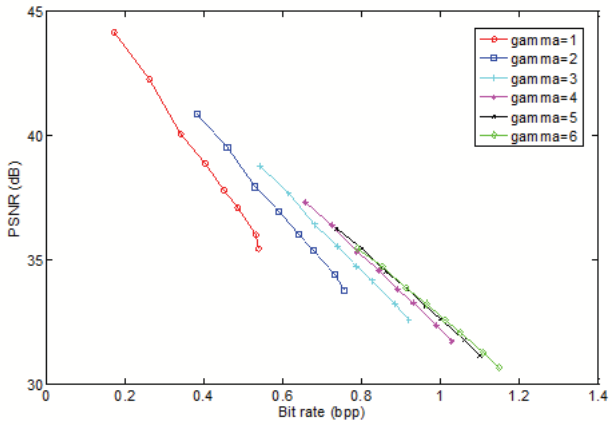


FIGURE 8. The relationship between PSNR and bit rate for the proposed method using a different set of γ and β.

two schemes. Moreover, if the mixed images manipulated by the third parties, the ones generated by our method are more robust than those generated by the other three schemes. Although the number of overhead information bits for our method is larger than that for the compared techniques, the overhead size of the proposed method can be reduced by the following two approaches. Firstly, using a single block-mean shared by a series of s-neighbor blocks. That is, to generate a reduced-size mean table that half the size of the original one, each element of the reduced-size table is obtained by computing the average of two consecutive elements in the original mean table. Then, two consecutive (hidden) blocks can share a single mean of the reduced-size mean table during data embedding (and extraction). A reduced-size mean table with a third or quarter of the original size can be generated in the same way. Secondly, a lossless arithmetic coding can be applied to the reduced-size mean table so as to further decrease the size of overhead information.

To demonstrate the robustness of the proposed method, examples of the survived watermarks including the bit correct ratio (BCR) are given in Fig. 9. The BCR is defined by

$$BCR = \frac{\sum_{i=0}^{MN-1} \overline{w_i \oplus \widehat{w}_i}}{M \times N} \times 100\% \tag{4}$$

where $w(i,j)$ and $\widehat{w}(i,j)$ represent the values of the original watermark and the extracted watermark, respectively. Although the BCR for the watermarks in Fig. 9(b) and 9(c) are a little bit low, the survived watermarks are recognized. In addition, the BCR of Fig. 9(e) is only 19.132%, the extracted watermark is recognized. It is interested that BCR of Fig. 9(f) is 100%, which means the mixed images generated by our method are immune from inverting attack. We therefore conclude that the mixed images introduced by the proposed method (using $\gamma = 1$ and $\beta = 8$) are tolerant from attcks such as brightness, JPEG2000 with a compression ratio (CR) of 1.56, JPEG with a CR=1.33, and inverting.

4. **Concluding Remarks.** In this paper, we present an effective reversible watermarking method based on the coefficient-bias algorithm. The proposed coefficient-bias algorithm consists of two phases. In phase I, the primary message with a large amount of bits is embedded in spatial domain to introduce a stego-image. To provide a better robustness performance, the secondary watermark with the smaller number of bits is subsequently

TABLE 1. The PSNR and bit rate generated by a variety of methods.

Image	Methods			
	Lin et al.'s algorithm [14]	Lin and Hsueh's technique [16]	Hsiao et al.'s approach [18]	Proposed method
Lena	30.2/0.88 (OH*=707)	30.0/1.18 (OH=692)	30.0/1.16 (OH>5,248)	30.00/1.20 (OH=226k)
Jet	30.1/1.10 (OH=703)	30.3/1.40 (OH=608)	30.0/1.09 (OH>7,936)	31.04/1.20 (OH=226k)
Peppers	30.2/0.91 (OH=29,977)	30.2/1.23 (OH=33,706)	30.0/1.16 (OH>10,624)	29.66/1.21 (OH=226k)
Goldhill	30.1/0.89 (OH=703)	30.1/1.16 (OH=845)	30.0/0.94 (OH>5,632)	29.65/0.94 (OH=226k)
Baboon	30.2/0.51 (OH=1,013)	30.4/0.61 (OH=1,426)	30.0/0.53 (OH>11,008)	30.48/0.43 (OH=226k)
Average	30.16/0.86 (OH=5,517)	30.2/1.12 (OH=6,213)	30.0/0.98 (OH>8,090)	30.17/1.00 (OH=226k)

OH*=OH stands for overhead bits.

hidden in the LH- and HL-subband of IWT domain, which transformed from a stego-image during the conduct of phase II. Experiments indicate that both the resultant perceptual quality and hiding capacity are not bad. Moreover, the mixed images generated by the proposed method do survive from various manipulations such as JPEG2000, JPEG, brightness, and inverting. Our future work will focus on the increment of payload size and the reduction of overhead bits.

(a) Attack-free(BCR=100%)

(b) Brightness (+40%) (BCR=70.560%)

(c) Brightness (-90%)(BCR=56.776%)

(d) JPEG2000 (CR=1.56)

(e) JPEG (CR=1.33)(BCR=19.132%)

(f) Inverting(BCR=100%)

FIGURE 9. Examples of extracted watermarks (of size 117×117 with 8 bits/pixel, 2-color) survived from the attacks.

REFERENCES

[1] J. S. Pan, H. C. Huang, and L. C. Jain, *Intelligent Watermarking Techniques*, Singapore: World Scientific, 2004.

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

[3] I. J. Cox, M. L. Miller, J. A. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, 2^{nd} Ed. , Morgan Kaufmann. , MA, 2008.

[4] C. K. Chan and L. M. Cheng, Hiding data in images by simple LSB substitution, *Pattern Recognition*, vol. 37, pp. 469-474, 2004.

[5] H. C. Wu, N. I. Wu, C. S. Tsai, and M. S. Hwang, Image steganographic scheme based on pixel-value differencing and LSB replacement methods, *IEEE Vision Image Signal Processing*, vol. 152, pp. 611-615, 2005.

[6] R. Z. Wang and Y. S. Chen, High-payload image steganography using two-way block matching, *IEEE Signal Processing Letters*, vol. 13, pp. 161-164, 2006.

[7] X. Zhu, A. T. S. Ho, and P. Marziliano, A new semi-fragile image watermarking with robust tampering restoration using irregular sampling, *Signal Processing: Image Communications*, vol. 22, pp. 515-528, 2007.

[8] S. Wang, B. Yang, and X. Niu, Secure steganography method based on genetic algorithm, *Journal of Information Hiding and Multimedia Signal Processing*, vol. 1, no. 1, pp. 28-35, 2010.

[9] T. H. Chen and T. H. Hung, Multiple watermarking based on visual secret sharing, *International Journal of Innovative Computing Information and Control*, vol. 4, no. 11, pp. 3027-3036, 2008.

[10] J. Tian, Reversible data embedding using a difference expansion, *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890-896, 2003.

[11] A. M. Alattar, Reversible watermark using the difference expansion of a generalized integer transform, *IEEE Trans. Image Processing*, vol. 13, no. 8, pp. 1147-1156, 2004.

[12] Z. Ni, Y. Q. Shi, N. Ansary, and W. Su, Reversible data hiding, *IEEE Trans. Circuit and System for Video Technology*, vol. 16, pp. 354-362, 2006.

[13] C. C. Chang, T. Lu, Y. F. Chang and C. T. Lee, Reversible data hiding schemes for deoxyribonucleic acid (DNA) medium, *International Journal of Innovative Computing Information and Control*, vol. 3, no.5, pp. 1145-1160, 2007.

[14] C. C. Lin, N. L. Hsueh, and W. H. Shen, High-performance reversible data hiding, *Fundamenta Informatica*, vol. 82, pp. 155-169, 2007.

[15] C. C. Chen and D. S. Kao, DCT-based zero replacement reversible image watermarking approach, *International Journal of Innovative Computing Information and Control*, vol. 4, no. 11, pp. 3027-3036, 2008.

[16] C. C. Lin and N. L. Hsueh, A lossless data hiding scheme based on three-pixel block differences, *Pattern Recognition*, vol. 41, pp. 1415-1425, 2008.

[17] H. W. Tseng and C. C. Chang, An extended difference expansion algorithm for reversible watermarking, *Image and Vision Computing*, vol. 26, pp. 1148-1153, 2008.

[18] J. Y. Hsiao, K. F. Chan, and J. M. Chang, Block-based reversible data embedding, *Signal Processing*, vol. 89, pp. 556-569, 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, pp. 1129-1143, 2009.