International Journal of Innovative Computing, Information and Control
Volume **11**, Number **5**, October **2015**

 ISSN 1349-4198
pp. 1715–1723

REVERSIBLE INFORMATION HIDING BASED ON HISTOGRAM SHIFTING FOR JPEG2000 IMAGES

Zhe-Ming Lu* and Xiang Li

School of Aeronautics and Astronautics
Zhejiang University
No. 38, Zheda Road, Hangzhou 310027, P. R. China
*Corresponding author: zheminglu@zju.edu.cn

Received February 2015; revised June 2015

Abstract. *As an important branch of information security, information hiding in images has been extensively studied in recent years. However, most schemes will introduce irreversible distortion to the original image. It is unallowable in some special applications such as legal imaging and medical imaging. So reversible information hiding deserves us to study. In addition, most existing information hiding schemes are for BMP images and JPEG images. There are few related schemes for JPEG2000 images. So it is very necessary for us to study reversible information hiding for JPEG2000 images. In this paper, we propose a reversible information hiding scheme for JPEG2000 compressed images. In the proposed scheme, high frequency subbands of the cover image are divided into blocks. In each block, the histogram generated by the quantized wavelet coefficients of the block is shifted to create gaps for embedding data. Experimental results show that the proposed scheme has satisfactory performance. Compared with other schemes, the proposed scheme has higher embedding capacity and better visual quality of the stego image.*
Keywords: JPEG2000, Reversible information hiding, Histogram shifting

1. **Introduction.** With the development of the Internet, information security is becoming more and more important. As an important branch of information security, information hiding technique has drawn a lot of attention in recent years. As is known to all, there are a lot of digital media files on the Internet. Information hiding can realize covert communication by embedding secret data into these media files. Among these media, digital image is a good and popular cover for information hiding. BMP images and JPEG images have been widely used for hiding data. JPEG2000 is a new international standard for still image compression, which is based on the wavelet transform and the embedded block coding with optimized truncation (EBCOT) algorithm [1,2]. JPEG2000 can provide superior compression performance over JPEG standard. In addition, it also provides a set of features such as region of interest (ROI) coding and progressive transmission. Due to its excellent coding performance and attractive features, JPEG2000 has a broad application prospect in the future. However, there are few related studies on information hiding for JPEG2000 images. So it is very necessary for us to study how to embed data into JPEG2000 images.

In order to protect secret information, information hiding in images has already been extensively studied. Many effective schemes have been proposed. Most of the existing information hiding schemes will introduce unrecoverable distortion to the original image, that is, the distortion cannot be removed after the hidden data are extracted. However, in some special applications such as legal imaging and medical imaging, it is critical to reverse the stego image back to the original image after the hidden data are extracted. Thus, most data hiding schemes such as the well-known least significant bit (LSB) algorithm cannot meet this requirement. So the concept of reversible information hiding or lossless data hiding was proposed. It can not only extract the hidden data, but also restore the cover image without any distortion. Since the idea was put forward, many reversible information hiding schemes have been proposed. These schemes can be broadly divided into three types: spatial domain based schemes, transform domain based schemes and compressed domain based schemes. In the spatial domain, there are many effective schemes. Ni et al. proposed a classic lossless data hiding scheme based on histogram shifting [3]. The scheme can embed data by shifting the image histogram. Leest et al. also proposed a lossless scheme based on histogram [4]. They utilize a compression function to introduce gaps for data hiding in the histogram. Compared with the former, this scheme can increase the embedding capacity. Tian proposed another classic reversible data hiding algorithm based on difference expansion [5]. The hidden data can be embedded by expanding the difference between two adjacent pixels. However, the embedded location map takes up much space. In addition, in recent years, Wu et al. proposed a novel data hiding algorithm with the property of contrast enhancement [6]. Luo et al. proposed a scheme based on hybrid prediction and inter-leaving histogram modification with single seed pixel recovery [7]. Shin and Jeon proposed a lossless data hiding technique using a reversible function and a pattern table [8]. In the transform domain, Yang et al. proposed a good reversible watermarking scheme [9]. This scheme takes advantage of integer discrete cosine transform coefficients' Laplacian-shape-like distribution and chooses AC coefficients for the bit shift operation. In the compressed domain, with regard to vector quantization (VQ) compressed images, Chu et al. proposed a high capacity reversible information hiding algorithm based on difference coding of VQ indices [10]. Zhao et al. proposed a high efficiency reversible data hiding scheme for two-stage VQ compressed images [11]. With regard to block truncation coding (BTC) compressed images, Li et al. proposed a reversible data hiding scheme based on bit-plane flipping and histogram shifting of mean tables [12]. With regard to JPEG compressed images, Hu et al. proposed a lossless data hiding scheme based on improved variable length codes (VLC) [13]. Jung proposed a new data hiding algorithm of embedding filter coefficients in JPEG bitstream [14]. With regard to JPEG2000 compressed images, Ohyama et al. proposed a lossless data hiding scheme using bit-depth information embedding [15].

In this paper, we propose a reversible information hiding scheme for JPEG2000 compressed images. Embedding data into JPEG2000 images has some differences from embedding data into other types of images. The differences will lead to some difficulties. As a compressed image, the redundancy of a JPEG2000 image is smaller than an uncompressed image such as a BMP image. So the space for data hiding is limited. This will increase the difficulty of data hiding. In addition, the encoding process of JPEG2000 is more complex. Some encoding operations such as quantization and bitstream layering probably destroy the hidden data. It requires that data hiding should be coordinate with the encoding process of JPEG2000. So selecting suitable embedding position is important and difficult. In order to realize reversible data hiding, our scheme utilizes histogram shifting technique to embed data into quantized wavelet coefficients in high frequency subbands of the cover image. Our shifting scheme is different from the scheme [3]. In reference [3], the histogram is generated from the pixel values in the image. However, in our scheme, the histogram is generated from the quantized wavelet coefficients in the small block. In addition, the way we shift the histogram is also different. In their scheme, the histogram is always shifting towards the zero point. However, in our scheme, we introduce a direction sign first. The value of the sign depends on the distribution of the histogram. Then the histogram of the block is shifted to the left or right adaptively according to the value of the direction sign. This can reduce the impact of histogram shifting on image visual quality to some extent. Moreover, the histogram will be shifted by three units in our work instead of one unit in their scheme. So every coefficient that is associated with the peak point can embed two bits instead of one bit. This can increase the embedding capacity. Experimental results demonstrate that the proposed scheme has satisfactory performance.

The rest of this paper is organized as follows. The related knowledge including JPEG 2000 codec, histogram shifting and the choice of embedding position is introduced in Section 2. The proposed scheme is presented in Section 3. Experimental results are given in Section 4. Finally, conclusions are drawn in Section 5.

2. **Related Knowledge.**

2.1. **JPEG2000 codec.** In order to understand the proposed scheme better, we introduce the JPEG2000 codec first. The block diagram of the JPEG2000 codec is shown in Figure 1. In the JPEG2000 encoder, image preprocessing is performed at first. Here, image preprocessing includes image tiling, DC level shifting and component transformation. Image tiling is applied to the original image by dividing it into rectangular non-overlapping blocks (tiles). These blocks can be processed independently. DC level shifting can ensure that the input sample data has a nominal dynamic range that is approximately centered about zero. Then component transformation or color transformation is performed on the tile-component data. Only two component transformations are defined in the baseline JPEG2000 codec, i.e., irreversible component transform (ICT) and reversible component transform (RCT). Both of them can map the image data from RGB color space to YCbCr color space. Next, these tiles are decomposed into different decomposition levels by the forward discrete wavelet transform (FDWT). After N levels of discrete wavelet transform, we can get the decomposed image which is made up of a low frequency subband called LL_N and $3 \times N$ high frequency subbands referred to as HL_i, LH_i, HH_i, where $i = 1, 2, \ldots, N$. Figure 2 shows the subband structure after three levels of discrete wavelet transform. Then the resulting wavelet coefficients in all subbands are quantized. Next, entropy coding including tier-1 coding and tier-2 coding is applied to the quantized blocks to generate JPEG2000 compressed bitstream. Up to this point, the encoding process is finished. The decoding process of the JPEG2000 codec is just the reverse of the encoding process.

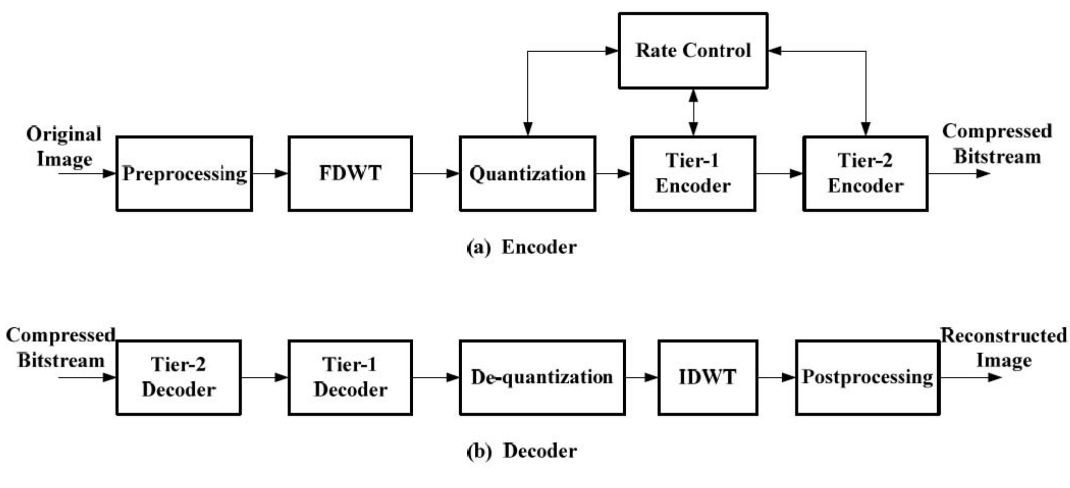

FIGURE 1. Block diagram of JPEG2000 codec

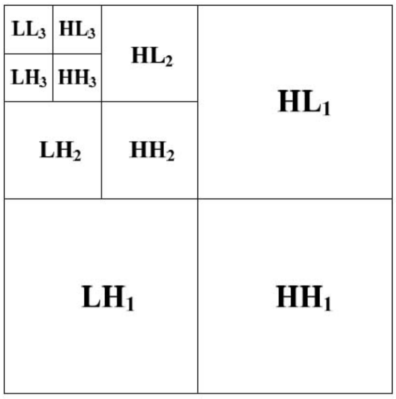

FIGURE 2. Subband structure

2.2. **Histogram shifting.** In digital image processing, histogram is a simple and practical tool. Ni et al. proposed a classic reversible data hiding scheme based on histogram shifting [3]. This scheme utilizes the peak and the zero or the minimum points of the image histogram and modifies the pixel values to embed secret data into the image. The peak point corresponds to the pixel value which has the maximum number of occurrences in the image. The zero point corresponds to the pixel value which has no occurrences in the image. In this scheme, the peak point and the zero point of the histogram are needed to be found first. Then the range between the peak point and the zero point of the histogram is shifted towards the zero point by one unit to create gaps for embedding data. Every pixel that is associated with the peak point can embed one bit. Obviously, the embedding capacity of this scheme is determined by the number of pixels that are associated with the peak point. In this case, the embedding capacity depends too much on the characteristics of the cover image itself. Aiming at the shortcomings of this algorithm, Leest et al. presented a scheme based on histogram [4]. In this scheme, a compression function is applied to the histogram to create space for data hiding. Compared with the former scheme, this scheme can increase the embedding capacity.

2.3. **The embedding position.** For information hiding, it is very important to select suitable embedding position. Due to the application of discrete wavelet transform (DWT) in JPEG2000 codec, a good choice is to embed secret data into the wavelet coefficients. However, next quantization stage will probably lead to a loss of hidden data. Thus, we consider embedding data into the quantized wavelet coefficients.

We have known that these coefficients are distributed in different subbands from the above introduction. Low frequency subband LL reflects the profile information of the image. Embedding data into the coefficients of this important subband has a big impact on the visual quality of the stego image. It will damage the invisibility of hidden data. On the contrary, high frequency subbands (HL, LH, HH) contain very little image information. Embedding data in these subbands has a little effect on image visual quality. So the quantized wavelet coefficients of high frequency subbands are better candidates for embedding data. In addition, we choose histogram shifting technique to embed data. For this technique, the embedding capacity of the scheme is determined by the height of the peak point in the histogram. For LL subband, the range of the quantized wavelet coefficients is very large, namely the coefficients are widely scattered. This feature will lead to a low height of the peak point. So the embedding capacity will be naturally low. On the contrary, the quantized wavelet coefficients in high frequency subbands (HL, LH, HH) are concentrated in a small range. So the height of the peak point is relatively higher.

The embedding capacity will be higher accordingly. For the above reasons, we decide to embed data into the quantized wavelet coefficients of high frequency subbands.

3. **The Proposed Scheme.** To embed data into a JPEG2000 image and recover the image without any distortion, the paper presents a reversible information hiding scheme based on histogram shifting. Our scheme is different from the existing scheme [15]. In their scheme, low frequency subband is also used for embedding data. However, we embed data into high frequency subbands only. In addition, they utilize the least significant bit plane of wavelet coefficients to embed data. However, we adopt different methods from them. We use a new histogram shifting method to create space for embedding data. In our scheme, we introduce a direction sign first. The value of the sign depends on the distribution of the local histogram. Then the local histogram is shifted to the left or right adaptively according to the direction sign. Every coefficient that is associated with the peak point of the histogram can embed two bits. The detailed procedure of the proposed scheme is described as follows.

3.1. **Data embedding.** The block diagram of data embedding is shown in Figure 3. In the embedding process, high frequency subbands of the cover image are divided into blocks. For each block, a new histogram shifting method is applied to the local histogram generated from all quantized wavelet coefficients in the block. The detailed embedding process is described as follows.

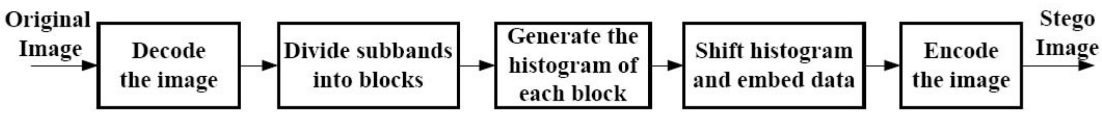

FIGURE 3. Block diagram of data embedding

Step 1: The original JPEG2000 image is partially decoded first. The decoding process stops before the de-quantization stage. So we can get the quantized wavelet coefficients of all subbands.

Step 2: Divide high frequency subbands into $m \times m$ blocks. The number of the blocks is denoted by N.

Step 3: For each block B_i $(i = 1, 2, \ldots, N)$, all coefficients in B_i are used to generate the local histogram of the block. Denote the peak point in the histogram as P_i. The number of times P_i occurring in the block is denoted by C_i. Then calculate T (the number of blocks that need to be used for embedding data) according to the number of hidden data bits and the values of C_i.

Step 4: For each needed block B_i $(i = 1, 2, \ldots, T)$, calculate L_i (the number of values that are distributed in the left side of P_i) and R_i (the number of values that are distributed in the right side of P_i). Define a direction sign S_i to mark the direction that the histogram shifts towards: $S_i = 1$ denotes the histogram shifts to the right, while $S_i = 0$ denotes the histogram shifts to the left. Then compare L_i with R_i. If $L_i \geq R_i$, $S_i = 1$. Otherwise, $S_i = 0$.

Step 5: If $S_i = 1$, shift the right side of P_i to the right by three units. It means that all coefficients in the right side of P_i are added by 3. If $S_i = 0$, shift the left side of P_i to the left by three units. It means that all coefficients in the left side of P_i are subtracted by 3. This step is used to create gaps for embedding data.

Step 6: Scan the coefficients of each block B_i $(i = 1, 2, \ldots, T)$ in a sequential order (row by row, from top to bottom). Once we encounter the coefficient whose value is P_i, check the hidden binary sequence and perform embedding operations until all data bits are embedded. There are four possible cases as follows.

Case 1: If the hidden bit string is "00", the coefficient remains unchanged.

Case 2: If the hidden bit string is "01", the coefficient is added by 1 while $S_i = 1$ or subtracted by 1 while $S_i = 0$.

Case 3: If the hidden bit string is "10", the coefficient is added by 2 while $S_i = 1$ or subtracted by 2 while $S_i = 0$.

Case 4: If the hidden bit string is "11", the coefficient is added by 3 while $S_i = 1$ or subtracted by 3 while $S_i = 0$.

Step 7: After all data bits have been embedded, perform encoding operations to generate the stego JPEG2000 image.

The values of P_i and S_i of block B_i $(i = 1, 2, \ldots, T)$ and the size of hidden data serve as a key. The sender can send the stego JPEG2000 image to the receiver in the public channel. Only getting the key can the receiver extract the hidden data from the image and recover the image.

3.2. **Data extraction and original image restoration.** The block diagram of data extraction and image restoration is shown in Figure 4. The detailed process can be described as follows.

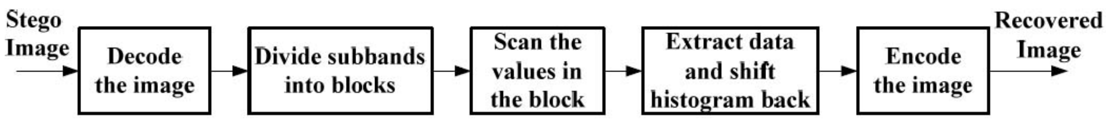

FIGURE 4. Block diagram of data extraction and image restoration

Step 1: At first, decode the stego JPEG2000 image partially. The decoding process stops before the de-quantization stage. So the modified quantized wavelet coefficients can be obtained.

Step 2: Divide high frequency subbands into $m \times m$ blocks. For each block B_i $(i = 1, 2, \ldots, T)$, scan the coefficients of the block in the same sequential order used in the above embedding process and perform corresponding operations according to the value of S_i. If all hidden data bits have been extracted, stop extracting data and just shift the histogram of the block back. The corresponding operations are described below.

(1) If $S_i = 1$, perform the following operations.

Scan the block. Once the coefficient between P_i and $P_i + 3$ is encountered (including P_i and $P_i + 3$), extract data from this coefficient. There are also four possible cases as follows.

Case 1: If the coefficient is equal to P_i, extract bit string "00". The coefficient remains unchanged.

Case 2: If the coefficient is equal to $P_i + 1$, extract bit string "01" and set the coefficient to P_i.

Case 3: If the coefficient is equal to $P_i + 2$, extract bit string "10" and set the coefficient to P_i.

Case 4: If the coefficient is equal to $P_i + 3$, extract bit string "11" and set the coefficient to P_i.

Scan the block again. If a coefficient is above $P_i + 3$, it will be subtracted by 3. The goal is to shift the histogram back. Other coefficients remain unchanged.

(2) If $S_i = 0$, perform the following operations.

Scan the block. If the coefficient is between $P_i - 3$ and P_i (including $P_i - 3$ and P_i), extract data from the value. In this case, P_i corresponds to "00", $P_i - 1$ corresponds to

"01", $P_i - 2$ corresponds to "10", and $P_i - 3$ corresponds to "11". After the bit string is extracted from the coefficient, the coefficient will be set to P_i.

Scan the block again. If a coefficient is below $P_i - 3$, it will be added by 3. The goal is to shift the histogram back. Other coefficients remain unchanged.

Step 3: After all data bits have been extracted, perform encoding operations to reconstruct the original image.

4. **Experimental Results.** In this section, some experimental results are given to evaluate the performance of the proposed scheme. The implementation of our scheme is based on JasPer [16] and Visual Studio 2010. The JasPer software is specified by the JPEG2000 standard. It can provide the implementation of the JPEG2000 codec. Some 512×512 JPEG2000 grayscale images are used as test images, as shown in Figure 5. The hidden data are a block of secret text. The text will be converted to a binary sequence. The number of levels of discrete wavelet transform we performed is five. In order to evaluate the similarity between the original image and the stego image, we choose their peak signal-to-noise ratio (PSNR) as the indicator. A larger PSNR value indicates a higher similarity and less image distortion. Generally, the stego image with 30dB PSNR or above will be regarded as visually acceptable.

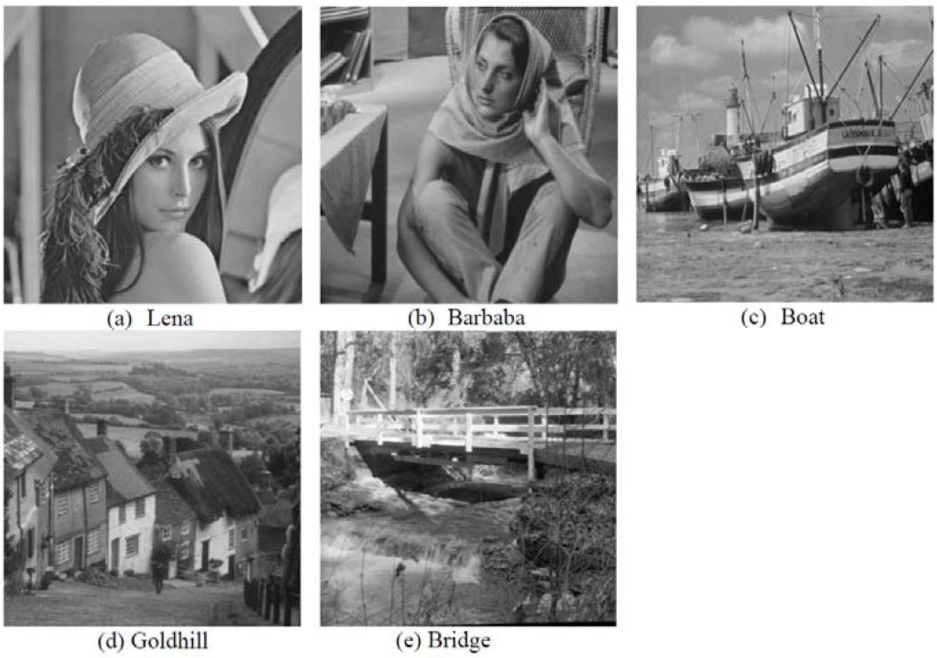

FIGURE 5. Five test images

In the proposed scheme, high frequency subbands are divided into $m \times m$ blocks. The choice of the block size is an important issue. So we need to analyze the relationship between the block size and the embedding capacity first. The block size is typically set to a power of 2. The experimental results are shown in Table 1.

From Table 1, we can find that the embedding capacity of the scheme decreases with the increase of block size on one hand. On the other hand, for convenience, the block size should not be too small. So a good choice is 8×8. In following experiments, the block size is 8×8.

For information hiding, the embedding capacity and the visual quality of stego image are both important indicators to measure the performance of the scheme. Table 2 shows

Table 1. The relationship between block size and embedding capacity

Image	Capacity (bit) (block size = 8 × 8)	Capacity (bit) (block size = 16 × 16)	Capacity (bit) (block size = 32 × 32)
Lena	95,172	82,052	77,288
Barbaba	79,662	67,084	61,948
Boat	66,824	53,788	48,454
Goldhill	68,246	55,038	49,632
Bridge	53,118	40,494	35,632

Table 2. PSNR values after embedding different amounts of data

Image	PSNR (dB) (0.5KB)	PSNR (dB) (1.0KB)	PSNR (dB) (1.5KB)
Lena	59.05	55.59	53.67
Barbaba	57.96	55.16	53.19
Boat	56.39	53.36	51.92
Goldhill	58.65	54.56	52.44
Bridge	55.62	53.07	51.59

Table 3. The results of the comparison

Image	Embedded secret data bits of our scheme (bit)	Embedded secret data bits of reference [15] (bit)
Lena	95,172	11,814
Barbaba	79,662	4,198
Bridge	53,118	9,502

Table 4. The comparison of PSNR

Image	Embedded bits (bit)	PSNR of our scheme (dB)	PSNR of reference [17] (dB)
Lena	1,024	66.03	45.23
Boat	1,024	62.76	44.80
Goldhill	1,024	65.83	43.90

the PSNR values of stego images after embedding different amounts of data. From Table 2, we can see that the proposed scheme achieves satisfactory PSNR values. In addition, in the experiments, the hidden data can be exactly extracted and the PSNR between the recovered image and the original image is infinite which means that the cover image can be totally restored.

Next, we compare our scheme with the scheme proposed by Ohyama et al. [15]. The results of the comparison are shown in Table 3. From the table we can see that our scheme can embed more secret data bits for the same image.

Then we further compare our scheme with the scheme proposed by Chen et al. [17]. The results of the comparison with their scheme are shown in Table 4. Compared with their scheme, we can see that our scheme has higher PSNR value after embedding equivalent amounts of data into the same image. It proves that our scheme has better invisibility of hidden data.

5. **Conclusions.** A reversible information hiding scheme for JPEG2000 images is proposed in this paper. In our scheme, high frequency subbands of the cover image are divided into blocks. In each block, a new histogram shifting method is applied to the histogram of the block for embedding data. Every coefficient that is associated with the peak point embeds two bits of data. In addition, we shift the histogram to the left or right adaptively according to a direction sign. Experimental results show that the proposed scheme has high embedding capacity and good invisibility of hidden data.

Acknowledgement. This work is partially supported by the National Natural Science Foundation of China under Grant No. 61171150. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] ISO/JEC FCD 15444-1, *JPEG2000 Part 1: Final Committee Draft Version1.0*, 2000.
[2] ISO/JEC FCD 15444-2, *JPEG2000 Part 2: Final Committee Draft*, 2000.
[3] Z. Ni, Y. Q. Shi, N. Ansari and W. Su, Reversible data hiding, *Proc. of 2003 International Symposium on Circuits and Systems*, Bangkok, Thailand, pp.912-915, 2003.
[4] A. Leest, M. Veen and F. Bruekers, Reversible image watermarking, *Proc. of 2003 International Conference on Image Processing*, vol.2, pp.731-734, 2003.
[5] 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.
[6] H. T. Wu, J. L. Dugelay and Y. Q. Shi, Reversible image data hiding with contrast enhancement, *IEEE Signal Processing Letters*, vol.22, no.1, pp.81-85, 2015.
[7] H. Luo, F. X. Yu, Z. L. Huang, H. Chen and Z. M. Lu, Reversible data hiding based on hybrid prediction and interleaving histogram modification with single seed pixel recovery, *Signal, Image and Video Processing*, vol.8, no.1, pp.813-818, 2014.
[8] S. H. Shin and J. C. Jeon, Lossless data hiding technique using reversible function, *International Journal of Security and Its Application*, vol.8, no.1, pp.389-400, 2014.
[9] B. Yang, M. Schmucker, W. Funk, C. Busch and S. Sun, Integer DCT-based reversible watermarking for image using companding technique, *Proc. of the SPIE*, vol.5306, pp.405-415, 2004.
[10] D. H. Chu, Z. M. Lu and J. X. Wang, A high capacity reversible information hiding algorithm based on difference coding of VQ indices, *ICIC Express Letters, Part B: Applications*, vol.3, no.4, pp.701-706, 2012.
[11] D. N. Zhao, W. X. Xie and Z. M. Lu, High efficiency reversible data hiding for two-stage vector quantization compressed images, *Journal of Information Hiding and Multimedia Signal Processing*, vol.5, no.4, pp.625-641, 2014.
[12] C. H. Li, Z. M. Lu and Y. X. Su, Reversible data hiding for BTC-compressed images based on bitplane flipping and histogram shifting of mean tables, *Information Technology Journal*, vol.10, no.3, pp.1421-1426, 2013.
[13] Y. J. Hu, K. Wang and Z. M. Lu, An improved VLC-based lossless data hiding scheme for JPEG images, *Journal of Systems and Software*, vol.86, no.8, pp.2166-2173, 2013.
[14] S. W. Jung, Adaptive post-filtering of JPEG compressed images considering compressed domain lossless data hiding, *Information Sciences*, vol.281, pp.355-364, 2014.
[15] S. Ohyama, M. Niimi, K. Yamawaki and H. Noda, Lossless data hiding using bit-depth embedding for JPEG2000 compressed bit-stream, *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp.151-154, 2008.
[16] *The JasPer Project*, http://www.ece.uvic.ca/∼mdadams/jasper/, 2008.
[17] T. S. Chen, J. Chen and J. G. Chen, A simple and efficient watermarking technique based on JPEG2000 codec, *Proc. of the IEEE 5th International Symposium on Multimedia Software Engineering*, pp.80-87, 2003.