



JPEG Reversible Data Hiding with Matrix Embedding

Fangjun Huang^(✉) and Jiayong Li

Guangdong Provincial Key Laboratory of Information Security Technology,
School of Data and Computer Science, Sun Yat-sen University,
Guangzhou 510006, GD, China
huangfj@mail.sysu.edu.cn, lijy393@foxmail.com

Abstract. Joint photographic experts group (JPEG) image is the most popular image format used in our daily life. Different from the reversible data hiding (RDH) in spatial domain image, JPEG RDH needs to consider not only the visual quality, but also the file storage size of the marked image. In this paper, we firstly implement the matrix embedding (which is originated from steganography) strategy into the field of JPEG RDH. Via considering the philosophy behind the JPEG encoder, those quantized discrete cosine transform (DCT) coefficients that may introduce less distortion are adaptively selected for modification in the embedding process. Experimental results demonstrate that higher visual quality and less storage size of the marked JPEG image can be obtained compared with the state-of-the-art methods.

Keywords: Reversible data hiding · JPEG · Distortion

1 Introduction

Reversible data hiding (RDH) is a discipline developed at the end of last century [1]. It can imperceptibly hide data into digital images, and more importantly, the original image can be reconstructed completely while extracting the embedded data.

The RDH algorithms have been proposed on the basis of three fundamental strategies: lossless compression (LC) [2, 3], difference expansion (DE) [4, 5], and histogram shifting (HS) [6, 7]. Recently, some HS based strategies, such as two-dimensional histogram modification [8], multiple histograms modification [9, 10], and optimized histogram modification [11–13] were proposed, which have been approaching the rate distortion (*i.e.*, the embedding rate vs. visual distortion) upper bound. However, those methods designed for uncompressed image cannot be applied to JPEG image directly, since they may lessen the visual quality or increase the storage size of the marked JPEG image unpredictably.

In [2], Fridrich and Goljan firstly proposed a JPEG RDH algorithm. With dividing some elements of the quantization table by an integer, space can be created for embedding information, which was further improved by Wang *et al.* [14]. These methods can preserve the visual quality of the marked image well. However, the storage size of the marked JPEG image may greatly increase. In [15], Mobasseri *et al.* proposed a JPEG RDH algorithm via modifying the Huffman table of the JPEG file,

which was further improved by Qian and Zhang [16], and Hu *et al.* [17]. These methods can preserve the visual quality and storage size of the JPEG file perfectly. However, the embedding capacity is limited, and this may restrict its application. In [18], Huang *et al.* extended the HS strategy from spatial domain into JPEG domain. Through expanding the quantized DCT (q DCT) coefficients with values 1 and -1 to carry message bits, and with a new adaptive selection (*i.e.*, block selection) strategy, a tradeoff between the embedding capacity and distortion (including visual quality and storage space of the marked image) can be obtained. Recently, some new adaptive selection (*i.e.*, block selection and frequency selection) strategies were proposed by Wedaj *et al.* [19] and Hou *et al.* [20].

In this paper, we present a new JPEG RDH algorithm with matrix embedding [21, 22]. As we know, matrix embedding is a technique initially proposed for in the field of steganography. Through using it, less alternation needs to be made to the cover image while embedding the same amount of information bits. However, matrix embedding is designed for secret communication and can only ensure the successful extraction of the embedded information bits. If it is utilized in RDH directly, the original carrier image cannot be restored after data hiding. In this paper, we firstly design a new approach to apply matrix embedding technology to JPEG RDH, which can restore the original carrier image completely while extracting the embedded message bits. Experimental results demonstrate that compared with the state-of-the-art algorithms, good visual quality can be easily obtained. Meanwhile, the storage size of the original JPEG file can be well preserved.

The remainder of this paper is organized as follows. In Sect. 2, the proposed matrix embedding based RDH scheme for JPEG image is introduced. Experimental results and the comparison study are discussed in Sect. 3. Finally, we conclude in Sect. 4.

2 Proposed Scheme

In this section, an overview of HS-based RDH is introduced first, and then, we give our proposed RDH scheme for JPEG images.

2.1 Introduction of HS-Based RDH

HS is the most widely used RDH approach nowadays. As introduced in [18], the HS-based JPEG RDH algorithm consists of four steps, which are shown in Fig. 1.

The first step is to generate a sequence with a small entropy from the host image, which is shown in Fig. 1(a). The sequence can be realized by using the original q DCT coefficient histogram, the difference histogram, or the prediction error histogram. As pointed out in [18], the original q DCT coefficient histogram of JPEG image is quite sharp, and the data hiding can be easily performed on the q DCT coefficient histogram.

The second step is to divide the obtained histogram into two regions, *i.e.*, inner region and outer region, which is shown in Fig. 1(b). In general, some higher bins are divided into the inner region, while the remaining bins are divided into the outer region. For example, in Fig. 1(b) the histogram bins corresponding to coefficients with value 0,

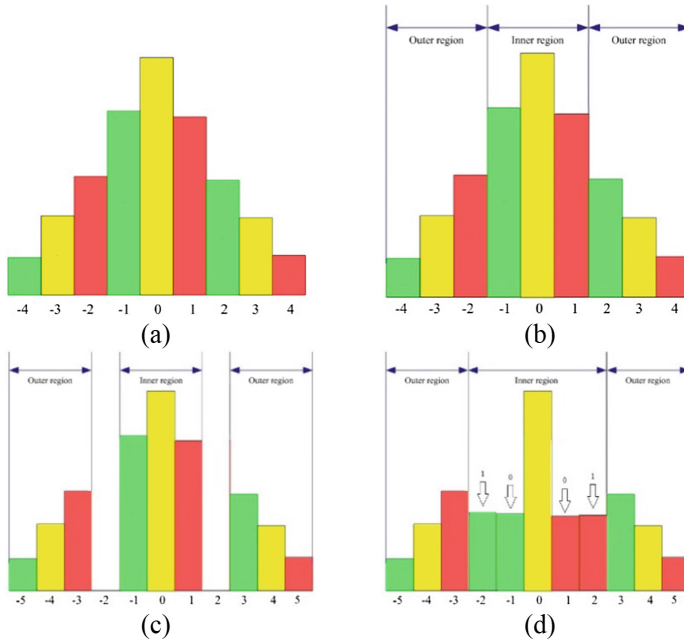


Fig. 1. The illustration of inner and out regions of HS method. (a) The original q DCT coefficient histogram of JPEG image, (b) the inner region and outer region, (c) shifting step, (d) expansion step.

1 and -1 are simply divided into inner region, and the remaining are divided into the outer region.

The third step is shifting, which is shown in Fig. 1(c). In this step, the bins in the outer region are shifted on both sides. Note that the shifting (and the subsequent expansion) of bins is achieved by modifying the corresponding q DCT coefficient values of the carrier image.

The fourth step is expansion, which is shown in Fig. 1(d). That is, the bins in the inner region are expanded to carry data. For example, for the coefficient with value 1, if message bit 1 is embedded, the coefficient becomes 2; otherwise if message bit 0 is embedded, the coefficient remains unchanged. Note that in this step, the coefficients with value 0 are not expanded to carry message bits.

As seen, in the third step the bins in the outer region are shifted to both sides, so that the inner and outer regions are still separate from each other after data hiding. In the receiving end, the receiver can easily extract the message bits and restore the original image.

2.2 Proposed JPEG RDH

As seen, for HS based JPEG RDH, the distortion mainly comes from two aspects, *i.e.*, shifting and expansion. Previously, the researchers mainly focused on how to reduce the number of shiftings and expansions, such as using more accurate prediction

algorithms to generate PEE histograms, and using adaptive selection strategies. However, for JPEG image, even though the two RDH methods have the same number of shiftings and expansions, they may result in different visual qualities and storage sizes due to the quantization, run length coding and Huffman coding in the compressing process. Thus locating those modification positions that may introduce less distortion is another important factor. As mentioned before, matrix embedding is a previously introduced technique for steganography to improve the embedding efficiency (*i.e.*, increase the number of bits embedded per embedding change). It can decrease the necessary number of changes. Furthermore, some modern matrix embedding strategy [22] can adaptively select those coefficients that may introduce less distortion for modification. Thus in this section, we will show how to apply matrix embedding to JPEG RDH.

Matrix Embedding. Suppose that the message sequence to be embedded is $\mathbf{m} = (m_1, \dots, m_p)^T$, where $m_i \in \{0, 1\}$. The least significant bits (LSBs) of the q DCT coefficients selected for data hiding is $\mathbf{x} = (x_1, \dots, x_q)^T$, where $x_i \in \{0, 1\}$ and $q \geq p$. It is assumed that the sender and recipient share a binary matrix A of dimension $p \times q$, which can be generated randomly or according to different matrix embedding strategies. The data hider will modify some bits in \mathbf{x} , so that the modified binary column $\mathbf{x}' = (x'_1, \dots, x'_q)^T$ satisfies:

$$A\mathbf{x}' = \mathbf{m} \quad (1)$$

Thus, the sender needs to solve a system of linear equations over Galois field GF(2). According to the law of matrix embedding, the rank of the generated matrix A should be equal to p in general. Since the binary matrix A is with the dimension $p \times q$ and $q > p$ generally, the above equation may have multiple solutions. Therefore, the sender can select the solution that may result in higher visual quality and less file size increments.

After data hiding, the embedded message should be extracted without error. With sharing a secret key with the sender, the recipient can extract the modified LSB sequence $\mathbf{x}' = (x'_1, \dots, x'_q)^T$, the decoding is very simple because the recipient can extract the embedded message according to Eq. (2) directly, *i.e.*,

$$\mathbf{m} = A\mathbf{x}' \quad (2)$$

However, the philosophy behind the matrix embedding can only ensure that the embedded message is extracted without error. As we know, in the field of RDH, the carrier image should also be restored completely. How to restore the carrier image will be explained in the next.

Distortion Function Designing. As introduced before, Eq. (1) may have many solutions, and the sender can select the solution that may result in minimal distortion. However, which solution can lead to minimal distortion?

Figure 2 illustrates the standard JPEG quantization table [23] corresponding to the quality factor (QF) of 80. It is known that the inverse discrete cosine transform (IDCT) is a linear transformation. Thus the modification to the q DCT coefficient associated with smaller quantization step may result in less visual distortion in the marked image [24, 25]. Based on this observation, in JPEG RDH, the message bits should be embedded into those coefficients belonging to the relatively low frequencies.

$$\begin{bmatrix} 6 & 4 & 4 & 6 & 10 & 16 & 20 & 24 \\ 5 & 5 & 6 & 8 & 10 & 23 & 24 & 22 \\ 6 & 5 & 6 & 10 & 16 & 23 & 28 & 22 \\ 6 & 7 & 9 & 12 & 20 & 35 & 32 & 25 \\ 7 & 9 & 15 & 22 & 27 & 44 & 41 & 31 \\ 10 & 14 & 22 & 26 & 32 & 42 & 45 & 37 \\ 20 & 26 & 31 & 35 & 41 & 48 & 48 & 40 \\ 29 & 37 & 38 & 39 & 45 & 40 & 41 & 40 \end{bmatrix}$$

Fig. 2. Standard JPEG quantization table corresponding to QF = 80.

Without loss of generality, the non-zero q DCT coefficients selected for RDH are represented by $\mathbf{c} = (c_1, c_2, \dots, c_N)$, where N represents the number of coefficients. Those selected coefficients are divided into two groups, *i.e.*, wet group (WG) and dry group (DG) [26]. In our method, all the coefficients with values 1 and -1 are divided into DG, and the rest are divide into WG. The coefficients in WG are assigned a large cost value and should be kept unchanged as many as possible; the coefficients in DG can be modified and assigned a cost value that is closely related to its associated quantization step. Suppose the quantization step associated with c_i is q_i . The distortion is represented as follows.

$$d_{c_i} = \begin{cases} q_i^\alpha & \text{if } c_i \in DG \\ \infty & \text{if } c_i \in WG \end{cases} \quad (3)$$

where α is a parameter that controls the impact of the quantization step. For all coefficients (c_1, c_2, \dots, c_N) , the distortion values $(d_{c_1}, d_{c_2}, \dots, d_{c_N})$ can be computed according to Eq. (3).

2.3 Embedding, Extraction, and Restoration

As before, the nonzero q DCT coefficients selected for RDH are represented by $\mathbf{c} = (c_1, c_2, \dots, c_N)$. Note that in our algorithm, the direct current (DC) coefficients are excluded, and only some non-zero alternating current (AC) coefficients are selected for matrix embedding. The number (N) of coefficients selected for RDH is determined by the number of message bits to be embedded and the distribution of AC coefficients.

Suppose the number of message bits is L , the number of coefficients with values 1 or -1 (*i.e.*, the dry coefficients) in the selected sequence \mathbf{c} should be no less than $\beta \times L (\beta \geq 1)$, where β is a control parameter. With an appropriately selected β value, the efficiency of the matrix embedding can be improved, whereas less wet coefficients

(i.e., the coefficients do not have values 1 and -1) need to be modified in the embedding process. For the selected coefficient sequence (note that how to select the coefficient sequence c will be introduced in Sect. 2.4), a coefficient histogram can be constructed. The Bin 1 and Bin -1 are grouped into the inner region, and the rest of the bins into the outer region. Our algorithm has the following four steps.

The first step is to shift all outer coefficients to both sides as follows.

$$\tilde{c}_i = \begin{cases} c_i & \text{if } |c_i| = 1 \\ c_i + \text{sign}(c_i) & \text{if } |c_i| > 1 \end{cases} \quad (4)$$

where

$$\text{sign}(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{if } x < 0 \end{cases} \quad (5)$$

After that, the shifted sequence $\tilde{c} = (\tilde{c}_1, \tilde{c}_2, \dots, \tilde{c}_N)$ can be obtained.

The second step is to embed the message with matrix embedding. Suppose the message sequence to be embedded is $\mathbf{m} = (m_1, m_2, \dots, m_L)$, where L represents the number of message bits to be embedded and $m_i \in \{0, 1\}$. The LSB sequence of the shifted sequence \tilde{c} is represented with $\mathbf{x} = (x_1, \dots, x_N)^T$, where $x_i \in \{0, 1\}$ and $N \geq L$. In our algorithm, the Syndrome-trellis Codes (STCs) [22] will be selected to embed the message bits because of its high embedding efficiency. According to our above analysis, when applying matrix embedding to the LSB sequence of those selected non-zero AC coefficients, the coefficients with minimal distortion, i.e., those dry coefficients in DG will be preferentially modified. Suppose that the modified LSB sequence corresponding to the minimal distortion is $\mathbf{y} = (y_1, \dots, y_N)$. The distortion is computed as follows.

$$D(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^N d_{c_i} f(x_i \neq y_i) \quad (6)$$

where

$$f(x) = \begin{cases} 1 & \text{if } x \text{ is true} \\ 0 & \text{if } x \text{ is false} \end{cases} \quad (7)$$

After achieving the optimal binary column \mathbf{y} , the shifted coefficient sequence \tilde{c} is modified as follows.

$$\tilde{\tilde{c}}_i = \begin{cases} \tilde{c}_i + \text{sign}(\tilde{c}_i) & \text{if } x_i \neq y_i \text{ and } |\tilde{c}_i| = 1 \\ \tilde{c}_i & \text{else} \end{cases} \quad (8)$$

In Eq. (8), the modified coefficient sequence is represented with $\tilde{\tilde{c}} = (\tilde{\tilde{c}}_1, \tilde{\tilde{c}}_2, \dots, \tilde{\tilde{c}}_N)$. Note that in Eq. (8), when $x_i = y_i$, no modification needs to be made. In the case of $x_i \neq y_i$

and $|\widetilde{C}_i| = 1$, the corresponding coefficient \widetilde{c}_i needs to be expanded to carry the message bits. Since $|\widetilde{C}_i| = 1$, the introduced distortion may be small according to our distortion definition in Eq. (3). In the case of $x_i \neq y_i$ and $|\widetilde{C}_i| \neq 1$, the LSB of corresponding coefficient \widetilde{c}_i may also need to be modified in the matrix embedding process, otherwise the embedded message may fail to be extracted correctly. However, according to Eq. (3), the coefficient not equal to 1 or -1 (*i.e.*, the wet coefficients) may introduce a large distortion while being modified. Thus in the case of $x_i \neq y_i$ and $|\widetilde{C}_i| \neq 1$, the coefficient will remain unchanged during the embedding process. But we will record the position of such coefficient, and the location will be embedded into some DC coefficients of the carrier image later as side information. In the experiments, we will demonstrate that with careful selection of the parameter β , few wet coefficients need to be modified in the embedding process.

In the extraction process, the side information are extracted first. Then locate the modified coefficient sequence $\widetilde{\mathbf{c}}$. Suppose the LSB sequence of $\widetilde{\mathbf{c}}$ is represented by $\widetilde{\mathbf{y}}_i = (\widetilde{y}_1, \widetilde{y}_2, \dots, \widetilde{y}_N)^T$. Note that according to the extracted side information, if there are some wet coefficients which need to be modified in the matrix embedding process (these coefficients are not really modified and only the locations are recorded), we need to flip those elements in the extraction process. After flipping, suppose the coefficient sequence is $\widetilde{\mathbf{y}} = (\widetilde{y}_1, \widetilde{y}_2, \dots, \widetilde{y}_q)$. The message extraction and image restoration can be described as follows:

$$m' = A\widetilde{\mathbf{y}}^T \tag{9}$$

$$c'_i = \begin{cases} \text{sign}(\widetilde{c}_i) & \text{if } 1 \leq |\widetilde{c}_i| \leq 2 \\ \widetilde{c}_i - \text{sign}(\widetilde{c}_i) & \text{if } |\widetilde{c}_i| \geq 3 \end{cases} \tag{10}$$

where $\widetilde{\mathbf{y}}^T$ represents the transpose of $\widetilde{\mathbf{y}}$, m' is the extracted message bit, and $\mathbf{c}' = (c'_1, c'_2, \dots, c'_N)$ is the restored quantized coefficient sequence, respectively.

2.4 Adaptive Selection Strategy

As shown, in our proposed method, the outer coefficients (with magnitude greater than 1) are shifted so that the inner and outer regions of the coefficient histogram are still kept separate from each other after data hiding. However, the inner coefficients are not expanded directly as that in the HS-based RDH scheme. As we know, if the number of the message bits to be embedded is L , in the HS-based RDH scheme, generally about $L/2$ inner coefficients need to be expanded since the message bits to be embedded are assumed to be uniformly distributed. When matrix embedding is adopted, the inner coefficients that need to be modified is generally less than $L/2$ because of the efficiency of the matrix embedding. That is the main reason why our new method may introduce

less distortion. Secondly, according to our analysis previously, when some adaptive matrix embedding strategy is adopted, those coefficients that may introduce less distortion (including visual quality and file size) will be chosen for modification preferentially. That is the second reason that our method may have better performance.

However, as discussed in [22], the efficiency of matrix embedding has a close relationship with the relative wetness and relative payload, where the relative wetness represents the number of wet coefficients (*i.e.*, the coefficients not with values 1 and -1) per those coefficients selected for matrix embedding, and the relative payload represents the number of bits to be embedded per those dry coefficients (*i.e.*, the coefficients with values 1 and -1) selected for matrix embedding. In order to make sure that the number of wet coefficients is as few as possible (though those wet coefficients are kept unchanged, the locations of some wet coefficients may need to be recorded in the matrix embedding process according to the introduction above), the relative wetness or relative payload should be decreased as low as possible. Moreover, the outer coefficients do not carry any information, but they need to be shifted; this invalid shifting may lead to worse visual quality and larger storage size of the marked image.

Thus, for our block selection strategy, the first goal is to find the 8×8 blocks with fewer outer coefficients, and the coefficients belonging to these coefficient blocks will take precedence for data hiding. As described in [18], the 8×8 blocks with more zero coefficients will have fewer outer coefficients in general. Moreover, even the two blocks have the same number of non-zero coefficients, they may result in different distortion because those non-zero coefficients may be associated with different quantization steps.

Suppose that in one 8×8 block B_i , the non-zero q DCT coefficients are represented with (NZ_1, \dots, NZ_K) , and the quantization steps associated with those coefficients are (Q_1, \dots, Q_K) , where $K \leq 63$. Thus for each block, the distortion value is represented by

$$D_{B_i} = Q_1^\gamma + Q_2^\gamma + \dots + Q_K^\gamma \quad (11)$$

where the parameter γ is selected as 2 in all our experiments. Those 8×8 blocks with less D_{B_i} will be chosen for RDH preferentially in our proposed algorithm.

Note that in the proposed algorithm, all the zero AC coefficients remain unchanged in the embedding process, and the distortion of each 8×8 block will never change in the data hiding process according to Eq. (11). The receiver can easily locate the blocks utilized for data hiding, then extract the embedded information and restore the host image.

2.5 Embedding, Extraction, and Restoration Steps

In the embedding process, side information such as message length, the parameter of STCs, the number of wet coefficients that need to be modified, and the positions of these wet coefficients, are embedded in the key determined l DC coefficients with LSB substitution. The original LSBs of those DC coefficients are appended as part of the payload before embedding. In the extraction and recovery process, the LSBs of the DC coefficients are read first to find the side information.

The embedding steps are as follows.

Step (1) Entropy-decode the original JPEG file to get the q DCT coefficients. Then, Compute d_{c_i} for each coefficient c_i according to Eq. (3), and calculate D_{B_i} according to Eq. (11) as well as the number of AC coefficients with values 1 and -1 for each 8×8 block B_i .

Step (2) Find the minimum threshold value T_D , and the corresponding 8×8 blocks whose distortion values computed according to Eq. (11) are no less than T_D . Note that the number of ± 1 coefficients (represented by S) belonging to those blocks should satisfy the following condition, *i.e.*, $S \geq \beta \times (L + l)$, where L denotes the number of message bits to be embedded, l represents the number of side information bits. For the ease of explanation, the nonzero AC coefficients in those blocks utilized for matrix embedding (ME) are called **ME coefficients** in the following.

Step (3) Scan all nonzero AC coefficients in a secret key determined order. If the coefficient to be visited is a **ME coefficient**, it will be shifted according to Eq. (4). When the number (represented by S) of visited coefficients is equal to $\beta \times (L + l)$, the scanning is completed.

Step (4) Extract all the LSBs of the **ME coefficients** visited in step 3, and embed the information bits (*i.e.*, the message bits to be embedded and the LSBs of the predetermined DC coefficients) with matrix embedding using the STCs [22]. The modified LSB sequence can be obtained, and the marked coefficient sequence can be computed according to Eq. (8). Note that in this step, all the wet coefficients are kept unchanged and the positions of those wet coefficients that may need to be modified in the matrix embedding process are recorded, which will be embedded into those predetermined DC coefficients in the next step.

Step (5) After all information bits (including the message bits and the LSBs of the predetermined DC coefficients) are embedded into the AC coefficients, the message length, the wet coefficient positions that may needed to be changed, and some other side information are embedded into the predetermined DC coefficients with LSB substitution.

Step (6) Entropy-encode the obtained coefficients to get the marked JPEG file.

The extraction and restoration steps are as follows.

Step (1) Entropy-decode the marked JPEG file to get the quantized DCT coefficients.

Step (2) Locate the predetermined DC coefficients, and extract the side information.

Step (3) Locate **ME coefficients** according to the key determined order, extract the information bits having been embedded and restore the original AC coefficients according to Eqs. (9) and (10), respectively.

Step (4) Restore the LSBs of the DC coefficients via using the extracted information bits.

Step (5) After all the q DCT (including DC and AC) coefficients are restored, entropy-encode the restored coefficients again to get the original JPEG file.

3 Experimental Results

In our experiments, the secret message bits are randomly generated, and the JPEG images are compressed with the optimized Huffman table by using the IJG toolbox [23]. The 1,000 images randomly selected from BOSSbase [27] are utilized in our testing.

To evaluate the performance of the proposed new scheme, the state-of-the-art RDH scheme [18] designed for JPEG images is selected for comparison. Note that, the main difference between our new proposed method and the scheme in [18] is the utilization of ME technology. The parameters regarding to ME are selected as $\alpha = 2$, $\beta = 1.25$, $\gamma = 2$ and $h = 16$, where α is a parameter to control the impact of the quantization step in Eq. (3), β is a parameter to control the message length, γ is a block selection parameter, and h regarding to the efficiency of ME [22]. In addition, in order to demonstrate the efficiency of the proposed block selection strategy, experimental results corresponding to the proposed algorithm without using the old block selection strategy [18] is also illustrated, which are represented with the label “Old Adaptive”.

Two aspects, namely visual quality and file size preservation, are discussed. The peak signal-to-noise ratio (PSNR) value, which is calculated between the original JPEG

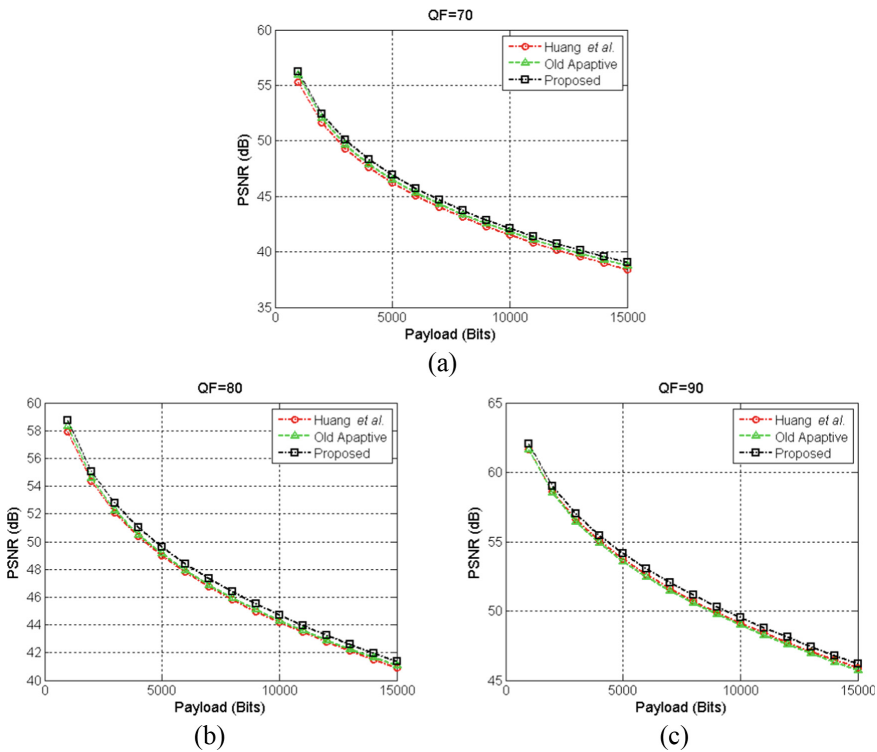


Fig. 3. Average PSNR values corresponding to different embedding payloads: (a) QF = 70 (b) QF = 80 (c) QF = 90.

image and the marked JPEG image, is used as a measure to evaluate the visual quality of the marked JPEG image.

3.1 Visual Quality

Three widely used image quality factors, *i.e.*, QF = 70, 80, 90 are tested. The testing results are shown in Fig. 3(a)–(c), where the horizontal axes represent the embedding payloads and the vertical axes represent the average PSNR values. The quality factor is shown in the title of each sub-figure. It is observed from Fig. 3 that the average PSNR values obtained by the proposed method are larger than those obtained by Huang *et al.*'s method [18] in general. However, without use of the block selection strategy (labeled as “Old adaptive”), the average PSNR values obtained by the proposed method will decrease, which implies the efficiency of our new adaptive selection strategy.

3.2 File Size Preservation

Figure 4 shows the average increased file sizes between the marked images and the original images. In Fig. 4(a)–(c), the horizontal axes represent the embedding payloads

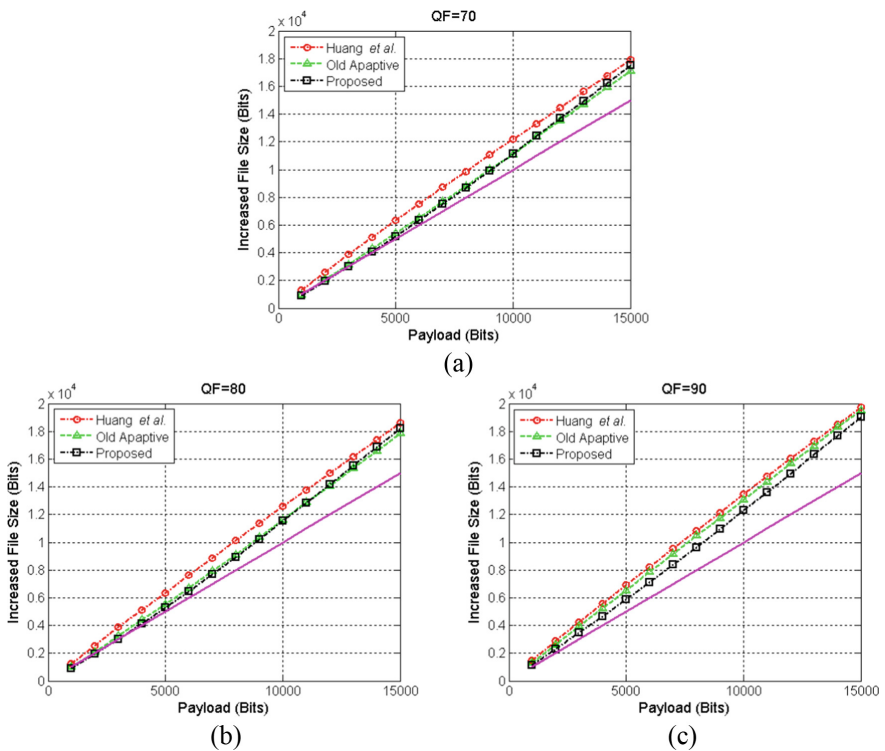


Fig. 4. Average increased file sizes corresponding to different embedding payloads: (a) QF = 70 (b) QF = 80 (c) QF = 90.

and the vertical axes represent the increased file sizes (bits). The solid magenta line in each sub-figure is used as a reference; every point on the line represents that the length of the embedded message bits is equal to the increased file size. As discussed earlier, the corresponding image quality factor is shown in the title of each sub-figure.

It is observed from Fig. 4 that the proposed method can preserve the file size constantly better than Huang *et al.*'s method [18]. Even if the block selection strategy is not applied in the embedding process, the increased file size of the proposed method is still less than that of Huang *et al.*'s methods. This implies that the proposed method can preserve the file size better than Huang *et al.*'s method.

4 Conclusions

Although a remarkable progress has been made in the field of JPEG RDH nowadays, compared to that in the airspace, there are still many shortcomings. Regardless of the visual quality or file size preservation, the performance of current methods is far from the upper bound. What's more, what is the upper bound for JPEG image, is still a problem. New JPEG RDH algorithms are called for. In this paper, we present a new RDH scheme for JPEG images. The main contributions of this paper are as follows:

- (1) We firstly introduce matrix embedding (which is originated from steganography) into JPEG RDH. Via using matrix embedding strategy, more message bits can be embedded with less modifications. Moreover, those q DCT coefficients that may introduce less distortion can be adaptively selected for modifications in the embedding process.
- (2) A novel block selection strategy has been proposed in this paper, which may result in better visual quality and less storage size of the marked JPEG file. It can also be utilized by other RDH schemes to improve their performance.
- (3) Experimental results demonstrate that higher visual quality and less storage size of the marked JPEG image can be obtained compared with the state-of-the-art methods.

Acknowledgements. This work is partially supported by the National Natural Science Foundation of China (61772572), the NSFC-NRF Scientific Cooperation Program (61811540409), and the Natural Science Foundation of Guangdong Province of China (2017A030313366).

References

1. Barton, J.M.: Method and apparatus for embedding authentication information within digital data. U.S. Patent 5646997 (1997)
2. Fridrich, J., Goljan, M.: Lossless data embedding for all image formats. In: SPIE Proceedings of Photonics West, Electronic Imaging, Security and Watermarking of Multimedia Contents, vol. 4675, pp. 572–583 (2002)
3. Celik, M., Sharma, G., Tekalp, A., Saber, E.: Lossless generalized-LSB data embedding. IEEE Trans. Image Process. **14**(2), 253–266 (2005)

4. Tian, J.: Reversible data embedding using a difference expansion. *IEEE Trans. Circuits Syst. Video Technol.* **13**(8), 890–896 (2003)
5. Alattar, A.M.: Reversible watermark using the difference expansion of a generalized integer transform. *IEEE Trans. Image Process.* **13**(8), 1147–1156 (2004)
6. Ni, Z., Shi, Y., Ansari, N., Wei, S.: Reversible data hiding. *IEEE Trans. Circuits Syst. Video Technol.* **16**(3), 354–362 (2006)
7. Li, X., Li, B., Yang, B., Zeng, T.: General framework to histogram-shifting-based reversible data hiding. *IEEE Trans. Image Process.* **22**(6), 2181–2191 (2013)
8. Ou, B., Li, X., Zhao, Y., Ni, R., Shi, Y.-Q.: Pairwise prediction-error expansion for efficient reversible data hiding. *IEEE Trans. Image Process.* **22**(12), 5010–5021 (2013)
9. Li, X., Zhang, W., Gui, X., Yang, B.: Efficient reversible data hiding based on multiple histograms modification. *IEEE Trans. Inf. Forensics Secur.* **10**(9), 2016–2027 (2015)
10. Wang, J., Ni, J., Zhang, X., Shi, Y.-Q.: Rate and distortion optimization for reversible data hiding using multiple histogram shifting. *IEEE Trans. Cybern.* **47**(2), 315–326 (2017)
11. Zhang, X.: Reversible data hiding with optimal value transfer. *IEEE Trans. Multimedia* **15**(2), 316–325 (2013)
12. Hu, X., Zhang, W., Li, X., Yu, N.: Minimum rate prediction and optimized histograms modification for reversible data hiding. *IEEE Trans. Inf. Forensics Secur.* **10**(3), 653–664 (2015)
13. Zhang, W., Hu, X., Li, X., Yu, N.: Optimal transition probability of reversible data hiding for general distortion metrics and its applications. *IEEE Trans. Image Process.* **24**(1), 294–304 (2015)
14. Wang, K., Lu, Z.-M., Hu, Y.-J.: A high capacity lossless data hiding scheme for JPEG images. *J. Syst. Software* **86**, 1965–1975 (2013)
15. Mobasser, B.G., Berger, R.J., Marcinak, M.P., NaikRaikar, Y.J.: Data embedding in JPEG bitstream by code mapping. *IEEE Trans. Image Process.* **19**(4), 958–966 (2010)
16. Qian, Z., Zhang, X.: Lossless data hiding in JPEG bitstream. *J. Syst. Softw.* **85**, 309–313 (2012)
17. Hu, Y., Wang, K., Lu, Z.-M.: An improved VLC-based lossless data hiding scheme for JPEG images. *J. Syst. Softw.* **86**, 2166–2173 (2013)
18. Huang, F., Qu, X., Kim, H.J., Huang, J.: Reversible data hiding in JPEG images. *IEEE Trans. Circuits Syst. Video Technol.* **26**(9), 1610–1621 (2016)
19. Wedaj, F.T., Kim, S., Kim, H.J., Huang, F.: Improved reversible data hiding in JPEG images based on new coefficient selection strategy. *EURASIP J. Image Video Process.* **63**, 1–11 (2017)
20. Hou, D., Wang, H., Zhang, W., Yu, N.: Reversible data hiding in JPEG image based on DCT frequency and block selection. *Signal Process.* **148**, 41–47 (2018)
21. Fridrich, J., Soukal, D.: Matrix embedding for Large Payloads. *IEEE Trans. Inf. Forensics Secur.* **1**(3), 390–395 (2006)
22. Filler, T., Judas, J., Fridrich, J.: Minimizing additive distortion in steganography using Syndrome-Trellis Codes. *IEEE Trans. Inf. Forensics Secur.* **6**(3), 920–935 (2010)
23. Wallace, G.K.: The JPEG still picture compression standard. *IEEE Trans. Consumer Electron.* **38**(1), xviii–xxxiv (1992)
24. Huang, F., Huang, J., Shi, Y.Q.: New channel selection rule for JPEG steganography. *IEEE Trans. Inf. Forensics Secur.* **7**(4), 1181–1191 (2012)
25. Huang, F., Kim, H.J.: Framework for improving the security performance of ordinary distortion functions of JPEG steganography. *Multimedia Tools Appl.* **75**, 281–296 (2016)
26. Fridrich, J., Goljan, M., Lisonek, P., Soukal, D.: Writing on wet paper. *IEEE Trans. Inf. Forensics Secur.* **53**(10), 3923–3935 (2005)
27. <http://www.agents.cz/boss/BOSSFinal/>