Yao et al. EURASIP Journal on Image and Video Processing (2018) 2018:41
https://doi.org/10.1186/s13640-018-0281-y

EURASIP Journal on Image and Video Processing

RESEARCH Open Access

A general framework for shiftable position-based dual-image reversible data hiding

Heng Yao[1,2], Chuan Qin[1], Zhenjun Tang[2*] and Xinpeng Zhang[3]

Abstract

This paper proposes an improved method for shiftable position-based dual-image reversible data hiding. During the procedure of embedding data, the total number of shiftable coordinates is set as the parameter to make a trade-off between distortion and embedding rate. First, the optimal parameter is sought and the one-to-one code table is generated according to the expected embedding rate. Then, the message data is embedded into the cover image through a simple and effective code table lookup scheme to generate two visually similar stego images. During the procedure of data extraction, the message can be extracted in a reverse manner. The experimental results demonstrate that the proposed method has higher PSNR (peak signal-to-noise ratio) values than our previous work in the case of high embedding rates.

Keywords: Reversible data hiding, Dual images, Shiftable positions

1 Introduction

Reversible data hiding (RDH) is a technique to embed message bits into a cover image in a reversible manner. During the phase of extracting the hidden data, all pixels of the original cover image can be recovered without any distortion, which is very important for military and medical applications. In the past two decades, RDH has received much attention and there are many work presented to promote the development of this research.

The first and most significant research direction is the RDH in uncompressed cover image. The cover image considered in this direction is uncompressed grayscale image in bitmap format. The mainstream of the current methods to hide data into an uncompressed image in a reversible manner can be roughly grouped into three aspects: difference expansion (DE) methods, histogram shift (HS) methods, and prediction-error expansion (PEE) methods. The first DE method was proposed by Tian [1], where integer wavelet transform was first conducted on the cover image to generate difference values, and then, the difference values were expanded to create the space for embedding the secret data. Weng et al. [2] improved the DE method by using the invariability of the sum of pixel pairs and pairwise difference adjustment (PDA) technique. The local-prediction-based DE method was proposed by Dragoi and Coltuc [3], where a predictor of least square was computed on a sub-block that centered on the pixel and the corresponding prediction error is expanded based on the framework of DE. The first HS method was proposed by Ni et al. [4], where the bins less than a given integer are shifted toward the left by one to create a vacant bin for data embedding before the message bits were subtracted from the values at the bin of the given integer. Luo et al. [5] improved the HS method by utilizing the difference between interpolation value and corresponding pixel value, i.e., interpolation error, as the predictor to generate a sharp prediction-error histogram (PEH). By using the partial differential equations, Qin et al. [6] proposed an inpainting-assisted predictor to generate a prediction image that has similar structural and geometric information as the cover image. To better exploit the spatial redundancy in natural images, amounts of PEE methods [7–11] were presented to improve the embedding efficiency. These PEE methods can be deemed as an integration of DE and HS methods. Sachnev et al. [7] used a rhombus predictor to estimate the center pixel before generating a PEH. Then, the bins in the center of the PEH were expanded to embed message bits, and the bilateral bins were shifted outwards to create the enough

* Correspondence: tangzj230@163.com
[2]Guangxi Key Lab of Multi-source Information Mining and Security, Guangxi Normal University, Guilin 541004, China
Full list of author information is available at the end of the article

vacancies. Through considering the correlations among prediction errors, Ou et al. [8] proposed a pairwise PEE method by jointing every two adjacent prediction errors and expanding the 2D PEH. Li et al. [9] proposed a multiple-histogram-based PEE method to further decrease the embedding distortion during the PEH expansion phase, where the multiple histograms were generated according to a local complexity measurement. Li et al. [10] proposed a pixel value ordering (PVO)-based PEE method, where all the pixel values in each sub-block were sorted and the pixel with the largest (or smallest) value was predicted by the second largest (or smallest) value. Weng et al. [11] improved PVO method through embedding data adaptively into each block according to a local complexity analysis. Apart from the above methods, there are some other research directions of RDH, such as RDH in encrypted domain [12–14], RDH for JPEG images [15, 16], RDH for RGB color images [17, 18], RDH with the capability of image integrity authentication [19, 20], RDH with image enhancement [21, 22], and RDH with dual-image mechanism [23–32].

As a branch of RDH research, dual-image RDH schemes have been well developed in recent years. Different from the single-image RDH technique, a pair of stego images is generated simultaneously in the procedure of data embedding for dual-image RDH, and in the delivery process, users cannot extract the message and restore the cover image unless they possess both stego images. In addition, due to the extremely high similarity between the two stego images, the embedding capacity is obviously higher than single-image RDH methods with the same distortion. Therefore, dual-image RDH is now attracting much more attention since it can provide the benefits of good security and high embedding capacity.

Several attempts to develop effective dual-image RDH have been made. The first dual-image RDH method was proposed in [23]. In this method, a 256×256 modulus matrix was first generated, and then, a set of 2-bit messages was converted into quinary symbols before combining a pair of quinary symbols as an indicator to modify the cover pixel. Finally, the values corresponding to the left and right diagonals in the matrix were sought according to the symbol set, and the values were determined as the final stego pixel pair. In [24], the embedding capacity was improved by transferring the binary message into decimal symbols and seeking the stego values along the right diagonal of the corresponding sub-blocks in the modulus matrix, so that the stego value for selection was expanded from 5 to 9. In [25], an asymmetric method was proposed to generate the first stego image through a classical modification direction method [26] and to generate the second stego image by conducting an analysis of the first stego image. In [27], two tentative stego images were generated through a least significant bit (LSB) matching method, and all pixels were classified into either reversible pixels or irreversible pixels with the former remaining unchanged and the latter being updated via a rule table. In another work [28], the dual image was first embedded by one bit simultaneously without using complex rules, and then, one image was treated as the prediction of the other image. The capacity was further improved by using the concept of prediction for embedding the secret bits. In [29], each pair of cover pixels was treated as the center point in the cross pattern, and then, the bits of the secret message were embedded by using a four-direction modification. To avoid irreversibility, a location-based identification was conducted to determine whether the present modification should be kept or whether its original value should be restored. In [30], the message bits were converted into quinary-based symbols without any under-utilization, and the symbols were embedded by controlling the relative positions of each pair of stego pixels. In [31], the center-folding strategy was used to reduce the value of the secret symbol, and the reduced symbols were embedded in dual stego images through a simple, but effective, averaging operation. Recently, in our previous work [32], the data to be hidden was embedded based on a strategy for selecting the coordinates of shiftable pixels with minimum distortion.

In this paper, we propose a novel code-mapping-based dual-image RDH method by using the maximum allowable position number as the parameter to make a trade-off between capacity and distortion, and we further integrate a general framework to select the optimal parameter adaptively. The rest of this paper is organized as follows. Section 2 and Section 3 introduce the motivation and concrete steps of the proposed method, respectively. Section 4 presents the experimental results, and Section 5 concludes the paper.

2 Motivation of the proposed method

In this section, the related work of dual-image RDH with center-folding strategy [31] and our previous work [32] are briefly introduced, and meanwhile, its potential improvement, i.e., the motivation of the proposed method, is also presented.

In [31], Lu et al. proposed a dual-image RDH method based on center-folding strategy. Assume that x, x_1, and x_2 are the original cover pixel and its corresponding marked dual pixel, respectively. First, an integer parameter, l, was set to control the embedding capacity. Note that l was greater than or equal to 2 in [31], and then, l binary to-be-embedded secret bits, denoted by $\{m_1, m_2, \ldots, m_l\}$, were combined as a set and transferred into a decimal value, d. The value of d ranged from 0 to $2^l - 1$. Next, d was converted to a smaller value, d',

which was within the range of $[-2^{l-1}, 2^{l-1}-1]$ by the center-folding strategy as

$$d' = d - 2^{l-1}. \tag{1}$$

Then, d' was embedded into x to generate a pair of stego pixels, i.e., x_1 and x_2, by the following averaging operation:

$$\begin{cases} x_1 = x + \text{floor}(d'/2) \\ x_2 = x - \text{ceil}(d'/2) \end{cases} \tag{2}$$

where functions floor and ceil indicate the rounding operation toward the directions of negative infinity and positive infinity, respectively. In the phase of message extraction and cover image recovery, x and d can be restored as

$$\begin{cases} x = \text{ceil}\left(\dfrac{x_1 + x_2}{2}\right) \\ d = d' + 2^{l-1} = (x_1 - x_2) + 2^{l-1} \end{cases} \tag{3}$$

Finally, original message bits $\{m_1, \ldots, m_k\}$ can be restored through a decimal-to-binary conversion of d.

Recently, an improved work of [31] was proposed by us in [32]. Let us revisit the problem of restoring x from x_1 and x_2 as referred to (3). First, a coordinate system of x_2 with respect to x_1 was set up. In this way, for a designated cover pixel x, the shiftable coordinate for (x_1, x_2) can be derived as being located at the pair straight line of $x_2 = -x_1 + 2x$ and $x_2 = -x_1 + 2x - 1$. Figure 1a shows a diagram of a coordinate system and the shiftable coordinates, and an enlarged view of its central region is also shown in Fig. 1b. It can be seen that the shiftable coordinates for embedding, in the case of $k = 3$, are (x, x), $(x-1, x)$, $(x, x-1)$, $(x-1, x+1)$, $(x+1, x-1)$, $(x-2, x+1)$, $(x+1, x-2)$, $(x-2, x+2)$, and $(x+2, x-2)$. To clarify the description, we allocated the position number, starting at 1 and increasing by increments of 1 to represent the shiftable coordinates. Note that, as the position number increases, the distortion of each pixel also increases. Due to the aim for all the RDH methods that generate less distortion with a fixed embedding rate, we set a parameter k in our previous work, which was an integer equal to or greater than 1 to make a trade-off between distortion and embedding rate. Following [31], the k-length message bits, $\{m_1, m_2, \ldots, m_k\}$, were converted into decimal number d, and different from [31], if d was equal to $2^k - 1$, an extra bit, m_{k+1}, was added to improve the capacity; d was updated as $d = d + m_{k+1}$ and then embedded into x through:

$$\begin{cases} x_1 = x + \text{floor}(d/4), x_2 = x_1 - (d/2), \text{if } d \text{ is an even number} \\ x_1 = x - \text{ceil}(d/4), x_2 = x_1 + \text{ceil}(d/2), \text{otherwise} \end{cases} \tag{4}$$

Eventually, the decimal number d can be restored in a reverse manner in the data extraction procedure.

Now, we analyze the embedding strategy of our previous work from a different perspective. Actually, for each embedding with parameter k, we generated a one-to-one code table to map the specific combination of k-length or $(k+1)$-length message bits, which is denoted by C, to its corresponding shiftable position number P. Note that the maximum position number for each k was $2^k + 1$. For the ease of intuitive comprehension, Table 1 lists the corresponding code table for $k = 2$ and $k = 3$. As seen in Table 1, the maximum position numbers for $k = 2$ and $k = 3$ are 5 and 9, respectively, and their average embedding bit lengths are 2.125 and 3.0625 bits, respectively. Indeed, according to our observations, the total number of allowable shift positions, denoted by n, is more significant and elaborate than parameter k in the practical embedding performance. Note that value of n can be set as any arbitrary odd number greater equal to or greater than 3, and when the values of k are low enough, such as $k = 1$ or 2, their corresponding n values are 3 or 5, respectively, and there is no gap between them. However, as the

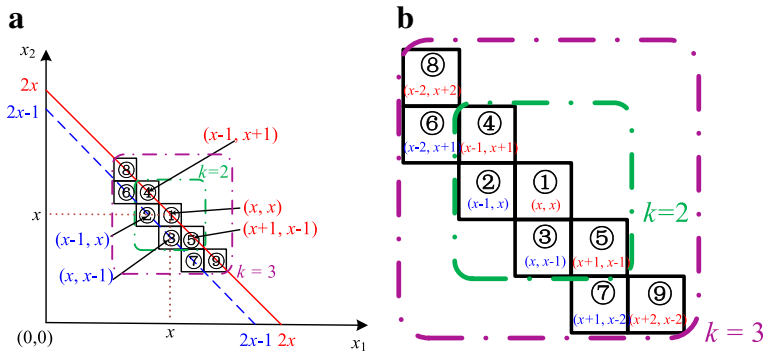

Fig. 1 Diagram of a coordinate system and allowable coordinates for the shiftable position-based dual-image RDH method. **a** Entire coordinate. **b** An enlarged view of central region of **a**

Table 1 One-to-one mapping code table for parameter k

$k = 2$			$k = 3$		
Code C	Position number P	Probability (%)	Code C	Position number P	Probability (%)
00	1	25	000	1	12.5
01	2	25	001	2	12.5
10	3	25	010	3	12.5
110	4	12.5	011	4	12.5
111	5	12.5	100	5	12.5
			101	6	12.5
			110	7	12.5
			1110	8	6.25
			1111	9	6.25

values of k increase, such as $k = 3$, 4, and 5, their corresponding n values are 9, 17, and 33. There are obvious gaps between 5 and 9, 9 and 17, and 17 and 33. For example, for $k = 2$ and 3, there is an optional setting of $n = 7$ between them, and similarly, for $k = 3$ and 4, there are triple optional settings of $n = 11$, 13, and 15 between them. Therefore, to improve the embedding efficiency, the code length parameter k is no longer of use in this paper; we now use the maximum position parameter n. The code table with respect to n can be generated in a similar manner as Table 1. Table 2 lists two examples, i.e., for $n = 7$ and 11. To better explain the key idea of the proposed method, an example of how to embed the message bits is shown in Fig. 2, where message bits $\{00\}_2$ and $\{101\}_2$ are embedded into the cover pixels 45 and 90, respectively. A more general framework is demonstrated in Section 3.

Table 2 One-to-one mapping code table for parameter n

$n = 7$			$n = 11$		
Code C	Position number P	Probability (%)	Code C	Position number P	Probability (%)
00	1	25	000	1	12.5
010	2	12.5	001	2	12.5
011	3	12.5	010	3	12.5
100	4	12.5	011	4	12.5
101	5	12.5	100	5	12.5
110	6	12.5	1010	6	6.25
111	7	12.5	1011	7	6.25
			1100	8	6.25
			1101	9	6.25
			1110	10	6.25
			1111	11	6.25

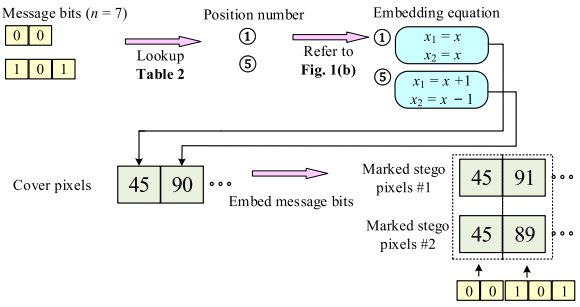

Fig. 2 An example of how to embed message bits for a given parameter n

3 Proposed method

Based on the observation that the maximum allowable position number n is more significant than code length parameter k used in [32], in this paper, we propose a novel code-mapping method to improve the embedding efficiency, especially when there is a requirement for a higher embedding rate. In addition, different from the previous methods [31, 32] that the parameters were selected manually, the optimal parameter n^* for a fixed embedding rate can be selected automatically before data hiding and n^* can be identified correctly in advance before data extraction. The proposed method consists of two procedures, i.e., embedding data and extracting data/image recovery. In addition, the issue of underflow/overflow also is addressed in this section. The detailed procedures are as follows.

3.1 Data embedding

For data embedding, the steps are as follows:

Step 1: Seek the optimal parameter n^*

In almost all dual-image reversible data hiding methods, the research objective is to obtain lower distortion with a fixed embedding rate α or embedding capacity β. The relationship between α and β can be derived by

$$\alpha = \frac{\beta}{M \times N \times 2} \tag{5}$$

where M and N are the row and column numbers of the cover image, respectively. According to (5), the optimization of embedding efficiency for a given β can be easily transferred to the optimization of that for a given α. Thus, in the remainder of this paper, we take the fixed embedding rate of α as an example. First, ignore the issue of underflow/overflow and set up the candidate parameter set $n = 3, 5, 7, 9, \ldots$, i.e., n can be set as an arbitrary odd integer greater than 1. As observed from Tables 1 and 2, the shortest code length, t, in the code table for each candidate n can be determined by

$$t = \text{floor}(\log_2(n)) \tag{6}$$

For instance, for $n = 7$, the shortest code is $\{00\}_2$ with the length of 2, and while for $n = 11$, the shortest codes are $\{000\}_2$, $\{001\}_2$, $\{010\}_2$, $\{011\}_2$, and $\{100\}_2$ with the length of 3. If t is given, the value of n belongs to the interval of $(2^t, 2^{t+1})$. In other words, once the value of n is fixed, for each embedding on each pixel, we can embed t to $t + 1$ binary bits. It can also be observed from Tables 1 and 2, assuming the message bits are randomly permutated and the numbers of 0 and 1 are approximately equal. Thus, the probability of each t-length code is $(2^{t+1} - n)/2^t$, and the probability of each $(t + 1)$-length code is $(n - 2^t)/2^t$. Also, take $n = 7$ and $n = 11$ for example; for $n = 7$, the probabilities of occurrence of $\{00\}_2$ and $\{010\}_2$ are 25 and 12.5%, respectively, and for $n = 11$, the probabilities of occurrence of $\{000\}_2$ and $\{1010\}_2$ are 12.5 and 6.25%, respectively. Based on the above observations, the ideal embedding rate α_n for any candidate n can be determined as

$$\alpha_n = \left[t \times \frac{2^{t+1} - n}{2^t} + (t+1) \times \frac{n - 2^t}{2^t} \right] \times \frac{1}{2} \tag{7}$$

Note that the reason for appearance of 1/2 in (7) is due to that the embedding rate is calculated according to the dual images rather than a single image.

Next, if we consider the problem of underflow/overflow, the number of pixels that cannot be modified should be identified. Here, we denote this number as K_n and we discuss how to determine K_n in Section 3.3. After determining K_n, the amendatory embedding rate for a fixed n can be updated as:

$$\tilde{\alpha}_n = \frac{(M \times N - K_n) \times \alpha_n}{M \times N} \tag{8}$$

For any candidate n, we calculate its limit-embedding rate $\tilde{\alpha}_n$, and the optimal parameter n^* can be determined by

$$n^* = \min(n) \quad \forall \tilde{\alpha}_n > \alpha, \quad n = 3, 5, 7, 9, \ldots \tag{9}$$

where min is the function to seek the minimum value.

Step 2: Generate the code table

The shiftable position number P is allocated from 1 to n^*, and the one-to-one mapping relationship between P and the binary code for embedding, i.e., C, should be established by the predefined regulations as described in Section 2. For mapping from code C to position number P, the first t^* bits are converted to decimal number d, where t^* is the shortest code length with respect to n^* and determined from (5). As can be observed from Tables 1 and 2, the number of codes is n^*, including the t^*-length codes and the $(t^* + 1)$-length codes with the numbers of $2^{t^*+1} - n^*$ and $2n^* - 2^{t^*+1}$, respectively. Then, the mapping relationship can be interpreted as:

$$\begin{cases} P = d + 1, & \text{if } d \le 2^{t^*+1} - n^* - 1 \\ P = \left(2^{t^*+1} - n^*\right) + 2 \times \left(d - 2^{t^*+1} + n^*\right) + m + 1, & \text{otherwise} \end{cases} \tag{10}$$

where m is the $(t^* + 1)^{th}$ bit of the message for embedding. For example, in the case of $n^* = 11$ and $C = \{001\}_2$, it can be easily calculated that $d = 1$, $t^* = 3$, and $d \le (2^{t^*+1} - n^* - 1)$; therefore, P can be directly determined by $P = d + 1 = 2$. If $n^* = 11$ and $C = \{1011\}_2$, in this case, $d = 5$, $t^* = 3$, and $d > (2^{t^*+1} - n^* - 1)$; therefore P can be determined by $P = (2^{t^*+1} - n^*) + 2 \times (d - 2^{t^*+1} + n^*) + m + 1 = 7$.

Step 3: Modify the cover pixel pair

Each original cover pixel, x, is modified to its corresponding marked stego pixel pair, x_1 and x_2, according to the parity of P as

$$\begin{cases} x_1 = x + \text{floor}\left(\frac{P-1}{4}\right), x_2 = x - \text{ceil}\left(\frac{P-1}{4}\right), & \text{if } P \text{ is an odd number} \\ x_1 = x - \text{ceil}\left(\frac{P}{4}\right), x_2 = x + \text{floor}\left(\frac{P}{4}\right), & \text{otherwise} \end{cases} \tag{11}$$

For example, as can be verified in Fig. 1, for $P = 2$, its corresponding modification equations are $x_1 = x - 1$, $x_2 = x$, and for $P = 7$, its corresponding modification equations are $x_1 = x + 1$, $x_2 = x - 2$.

To this point, the message bits have been embedded into cover image with a dual-image mechanism.

3.2 Extraction of data and restoration of the cover image

To extract the message and restore the cover image, the steps are listed as follows:

Step 1: Restore the cover image and identify n^*

For each pixel, the cover image can be restored through (3). Based on all groups of $\{x, x_1 \text{ and } x_2\}$, n^* can be identified as

$$n^* = \max(|x_1 - x| + |x_2 - x|) \times 2 + 1, \forall \{x, x_1, x_2\} \tag{12}$$

where max is the function to seek the maximum value.

Step 2: Regenerate the code table

In the extraction procedure, the code table is established with the inverse order relative to the embedding procedure. First, after recognizing the unmodified pixels with the consideration of underflow/overflow, the position number PP for the remaining pixels can be identified as

$$P = \begin{cases} (|x_1 - x| + |x_2 - x|) \times 2 + 1, & \text{if } x_2 \le x_1 \\ (|x_1 - x| + |x_2 - x|) \times 2, & \text{otherwise} \end{cases} \tag{13}$$

Next, establish the one-to-one mapping relationship from P Pto code C through

$$C = \begin{cases} \{P-1\}_{2|t^*}, & \text{if } P \le 2^{t^*+1}-n^* \\ \left\{ \text{ceil}\left(\frac{P-(2^{t^*+1}-n^*)}{2}\right) + 2^{t^*+1}-n^*-1 \right\}_{2|t^*} \Big\| \{ \text{mod}(P,2)\}_{2|1}, & \text{otherwise} \end{cases} \tag{14}$$

where, $\{\cdot\}_{2|t^*}$ is the conversion from decimal number to t^*-length binary digits, and $|\cdot|$ is a cascading operation onto two binary sequences.

Step 3: Extract the message bits

For each pixel without any underflow/overflow protection, after identifying its position number P and its corresponding code C, the message bits from each pixel can be extracted separately according to the code table as regenerated in Step 2. Finally, concatenate all messages into the entire message.

3.3 Underflow/overflow issue in the proposed method

For all RDH methods, the issue of underflow/overflow is inevitable. To avoid already modified pixels out of the bound of image representation, some original pixels are protected without any modification in the embedding process. The protection scope is

$$\begin{cases} \text{underflow} : x-\text{ceil}((n-1)/4) < 0 \\ \text{overflow} : x+\text{floor}((n-1)/4) > 255 \end{cases} \tag{15}$$

Thus, pixels that belong to the intervals of [0, ceil((*n*−1)/4)⌈(*n* − 1)/4⌉) and (255 − floor((*n* − 1)/4)⌈(*n* − 1)/4⌉)⌊(*n* − 1)/4⌋, 255] are not embedded in any message; instead, all remaining pixels are modified following the procedures described in Section 3.1. Note that the total number of pixels that belong to the underflow/overflow protection ranges is counted and denoted by K_n.

4 Results and discussion

To demonstrate the efficacy of the proposed method, we compared our method with some state-of-the-art dual-image RDH methods [28, 31, 32]. First, four standard test images, i.e., Lena, Baboon, Barbara, and Goldhill, were used in our evaluation as shown in Fig. 3.

Figure 4 shows the comparative results of the average peak signal-to-noise ratio (PSNR) with respect to the embedding rate between our previous work [32], Lu et al.'s method [31], and the proposed method. Note that owing to being very close to the corresponding PSNR performances of [32] in the cases of $k \ge 4$, for ease of comparison, the performance curves of [31] in the cases of $l \ge 4$ are omitted in Fig. 4. Note that, for shiftable position-based, dual-image RDH methods, their corresponding embedding performances were essentially unaffected by the content of the cover image itself, except for small differences in underflow/overflow. Therefore, as shown in Fig. 4, there is no significant difference between each image by using a specified method, whether it is [32] or the proposed method. It also should be pointed out that the parameters k and l in [31, 32], respectively, were not selected automatically; thus, there are multiple curves in each performance diagram based on the different settings of k and l. For the proposed method, the optimal parameter n^* can be sought adaptively, so the performance curve for the proposed method is integrated into a single curve. Due to the use of new parameter n^*, the performance of our method provided an obvious improvement, especially for the higher embedding rates.

To further demonstrate the superiority for the requirement of higher embedding rates, Table 3 lists the average PSNRs of the proposed method and some state-of-the-art methods, such as [28, 31, 32], with some fixed embedding rates. Due to the limited capacity for [28], we use "n/a" to stand for the occasion that is unavailable to fulfill the required capacity. Note that for [31, 32], based on the given embedding rates, we select the optimal parameter manually. It can be seen that all average PSNRs by the proposed method are higher than our previous work of

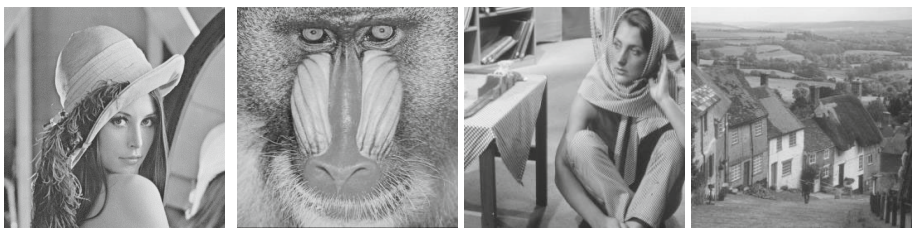

Fig. 3 Four standard test images with sizes of 512 × 512

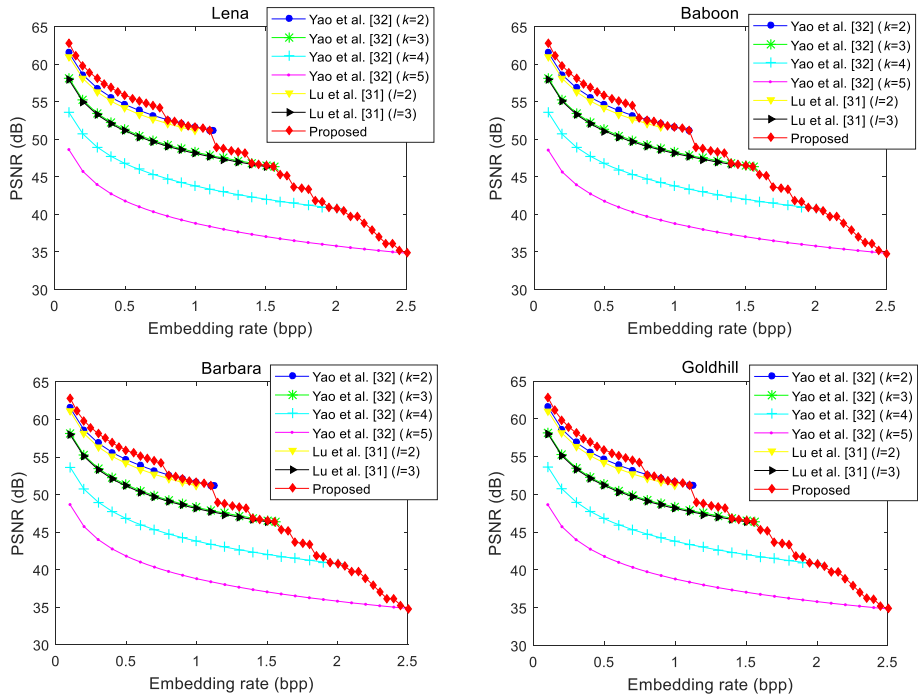

Fig. 4 Comparison of embedding rate and average PSNR between our previous method [32], Lu et al.'s method [31], and the proposed method

[32] and Lu et al.'s method [31], and in the case of $\alpha = 1.2$, average PSNRs in [28] is slightly higher than the proposed method. However, with the increase of α, the embedding rate is out of the maximum sustainable rate; in other words, the proposed method has more flexible capacity than [28].

Next, to demonstrate the universality of the proposed method, i.e., our method was not only effective on some specific standard test images but also arbitrary selected images, 300 images were randomly selected from UCID dataset to involve into the test after a conversion from RGB color channels to gray images. Figure 5 shows the average PSNRs of each dual image pair with two fixed embedding rates of 1.2 bpp (bits per pixel) and 1.6 bpp, respectively. To present the superiority of the proposed method, the corresponding average PSNRs of [32] with the same embedding rates are also shown in Fig. 5. It can be seen that the average PSNRs of the proposed method for the embedding rates of 1.2 bpp and 1.6 bpp were 48.71 and 45.07 dB, respectively, with the average gains of 1.19 and 3.38 dB, respectively, over our previous method [32].

Table 3 Comparison of the proposed method with other methods in terms of average PSNRs (dB) with fixed embedding rates (bpp). The best result for each image is presented in italics

Image	Embedding rate α	Lu et al. [31]	Yao et al. [32]	Jafar et al. [28]	Proposed
Lena	1.20	47.33	47.51	*48.75*	48.71
	1.60	41.70	41.75	n/a	*45.34*
	2.20	35.35	35.40	n/a	*38.86*
Baboon	1.20	47.33	47.52	*48.74*	48.72
	1.60	41.70	41.75	n/a	*45.34*
	2.20	35.36	35.39	n/a	*38.85*
Barbara	1.20	47.34	47.52	*48.74*	48.72
	1.60	41.70	41.78	n/a	*45.33*
	2.20	35.36	35.40	n/a	*38.86*
Goldhill	1.20	47.33	47.51	*48.74*	48.71
	1.60	41.70	41.75	n/a	*45.34*
	2.20	35.34	35.38	n/a	*38.87*

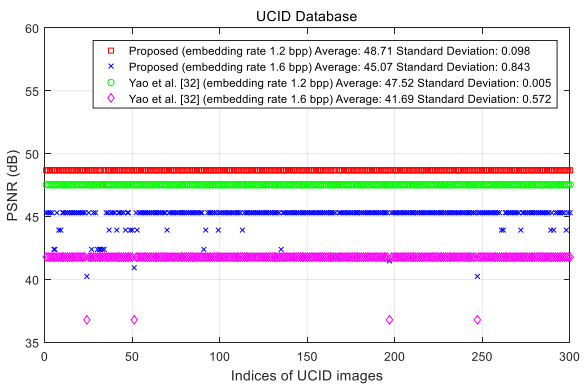

Fig. 5 Comparison of average PSNR for the specified embedding rates of 1.2 bpp and 1.6 bpp between the proposed method and previous method [32] using randomly selected 300 UCID images

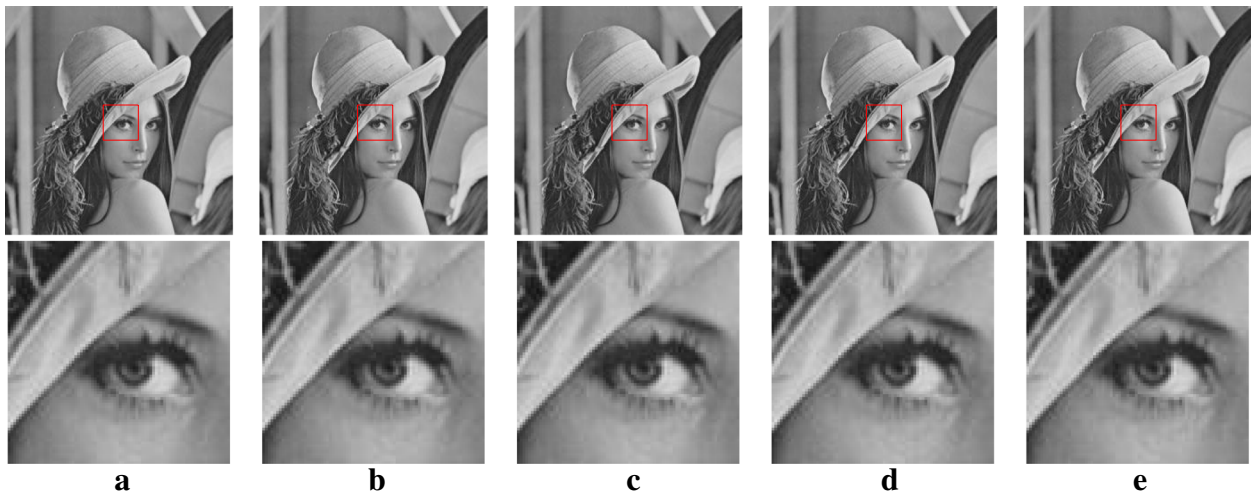

Fig. 6 Visual comparison of the cover image and the marked stego dual images. **a** Cover image. **b**, **c** Marked stego dual images with the embedding rate of 1.2. **e**, **f** Marked stego dual images with the embedding rate of 1.6. All bottom images are the partial enlarged view of the red square regions from their corresponding top images

Standard image Lena is adopted to demonstrate the imperceptibility of the proposed method. Figure 6 shows the comparison between the original cover image and the marked stego images, where the column (a) is the cover image and the zoom-in view of its red square region, the columns (b) and (c) are the marked dual images with the embedding rate of 1.2 bpp and their corresponding zoom-in versions, the columns (e) and (f) are the marked dual images with the embedding rate of 1.6 bpp and their corresponding zoom-in versions. As seen in Fig. 6, the marked dual images still exhibit acceptable visual quality even with a relative higher embedding rate.

To verify the security of the proposed algorithm, the attacks of RS steganalysis [33] is conducted on the proposed method. The RS steganalysis was designed to detect the trace of LSB steganography and estimate the embedding rate of hidden message. In our experiments, the RS test was conducted on the two test images, i.e., Lena and Barbara, respectively, with the different embedding rates. Figure 7 shows the estimated steganography rate with respect to the actual embedding rates varying from 0 to 2.0 with the incremental step of 0.1. Note that embedding rate = 0 means the original cover image without any distortion. As can be seen in Fig. 7, all estimated steganography rates of the marked images are below 10%, which is a relative safe threshold to avoid causing the alert by the attackers. Furthermore, the histogram analysis is also applied on the marked dual images. Figure 8 shows the histogram comparison of the cover images, Lena and Barbara, and their corresponding marked stego dual images with the embedding rate of 1.2. It can be seen in Fig. 8 that there is no obvious differences between the cover images and the marked images; in other words, the proposed method is with high security to resist the histogram analysis attack.

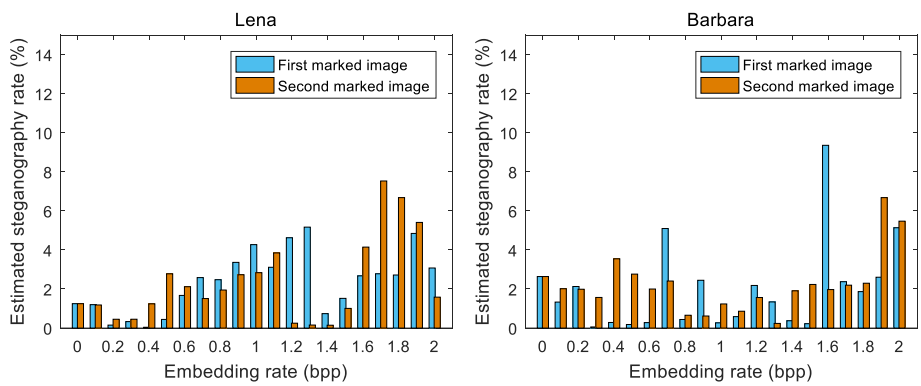

Fig. 7 RS steganalysis results of the proposed method with different embedding rates

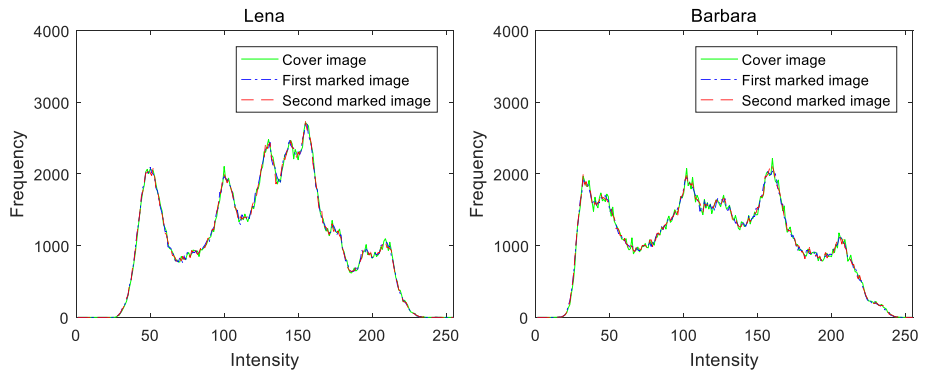

Fig. 8 Histogram comparison of the cover images and their corresponding marked dual images with the embedding rate of 1.2

5 Conclusions

In this paper, we have proposed a general work for our previously proposed shiftable position-based, dual-image RDH method. Compared with the previous method, the proposed method used the total number of shiftable positions, rather than the code length, as the embedding parameter. The proposed method provides two main improvements, i.e., (1) the optimal parameter can be sought automatically with a designated embedding rate and (2) the efficiency in the case of high embedding rates outperforms our previous work. The experimental results have demonstrated the improvements provided by the proposed method.

Abbreviations

bpp: Bits per pixel; DE: Difference expansion; HS: Histogram shift; LSB: Least significant bit; PDA: Pairwise difference adjustment; PEE: Prediction-error expansion; PEH: Prediction-error histogram; PSNR: Peak signal-to-noise ratio; PVO: Pixel value ordering; RDH: Reversible data hiding

Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful comments.

Funding

This work was supported in part by the National Natural Science Foundation of China (61702332, 61672354, 61562007, U1636206, 61525203, and 61472235), Research Fund of Guangxi Key Lab of Multi-source Information Mining & Security (MIMS16-03), the Guangxi Natural Science Foundation (2017GXNSFAA198222), the Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing, the Shanghai Dawn Scholar Plan (14SG36), and the Shanghai Excellent Academic Leader Plan (16XD1401200).

Authors' contributions

HY did the main work. CQ and ZT did the experiments. XZ offered the suggestion. All authors read and approved the final manuscript.

Authors' information

Heng Yao received the B.S. degree from Hefei University of Technology, China, in 2004; the M.S. degree from Shanghai Normal University, China, in 2008; and the Ph.D. degree from Shanghai University, China, in 2012. Currently, he is with School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, China. His research interests include digital forensics, data hiding, image processing, and pattern recognition.

Chuan Qin received the B.S. degree in electronic engineering and the M.S. degree in signal and information processing from Hefei University of Technology, Anhui, China, in 2002 and 2005, respectively, and the Ph.D. degree in signal and information processing from Shanghai University, Shanghai, China, in 2008. Since 2008, he has been with the faculty of the School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, where he is currently an Associate Professor. He was with Feng Chia University at Taiwan as a Postdoctoral Researcher and Adjunct Assistant Professor from July 2010 to July 2012. His research interests include image processing and multimedia security. He has published more than 90 papers in these research areas.

Zhenjun Tang received the B. S. and M. Eng. degrees from Guangxi Normal University, Guilin, P.R. China, in 2003 and 2006, respectively, and the Ph.D. degree from Shanghai University, Shanghai, P.R. China, in 2010. He is now a Professor with the Department of Computer Science, Guangxi Normal University. His research interests include image processing and multimedia security. He has contributed more than 50 international journal papers. He holds six China patents. He is a senior member of China Computer Federation (CCF) and also a reviewer of more than 20 SCI journals, such as IEEE journals, IET journals, Elsevier journals, Springer journals, and Taylor & Francis journals.

Xinpeng Zhang received the B.S. degree in computational mathematics from Jilin University, China, in 1995 and the M.E. and Ph.D. degrees in communication and information system from Shanghai University, China, in 2001 and 2004, respectively. Since 2004, he has been with the faculty of the School of Communication and Information Engineering, Shanghai University, where he is currently a Professor. He was with the State University of New York at Binghamton as a visiting scholar from January 2010 to January 2011 and with University of Konstanz as an experienced researcher sponsored by the Alexander von Humboldt Foundation from May 2011 to August 2012. His research interests include multimedia security, image processing, and digital forensics.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Author details

[1]Shanghai Key Lab of Modern Optical System, and Engineering Research Center of Optical Instrument and System, Ministry of Education, University of Shanghai for Science and Technology, Shanghai 200093, China. [2]Guangxi Key Lab of Multi-source Information Mining and Security, Guangxi Normal University, Guilin 541004, China. [3]School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China.

Received: 22 August 2017 Accepted: 22 May 2018
Published online: 06 June 2018

References

1. J Tian, Reversible data embedding using a difference expansion. IEEE Trans. Circuits Syst. Video Technol. **13**(8), 890–896 (2003)
2. S Weng, Y Zhao, JS Pan, R Ni, Reversible watermarking based on invariability and adjustment on pixel pairs. IEEE Signal Process Lett. **15**, 721–724 (2008)
3. I Dragoi, D Coltuc, Local-prediction-based difference expansion reversible watermarking. IEEE Trans. Image Process. **23**(4), 1779–1790 (2014)
4. Z Ni, Y Shi, N Ansari, W Su, Reversible data hiding. IEEE Trans. Circuits Syst. Video Technol. **16**(3), 354–362 (2006)
5. L Luo, Z Chen, M Chen, X Zeng, Z Xiong, Reversible image watermarking using interpolation technique. IEEE Trans. Inf. Forensics Secur. **5**(1), 187–193 (2010)
6. C Qin, CC Chang, YH Huang, LT Liao, An inpainting-assisted reversible steganographic scheme using a histogram shifting mechanism. IEEE Trans. Circuits Syst. Video Technol. **23**(7), 1109–1118 (2013)
7. V Sachnev, HJ Kim, J Nam, S Suresh, Y Shi, Reversible watermarking algorithm using sorting and prediction. IEEE Trans. Circuits Syst. Video Technol. **19**(7), 989–999 (2009)
8. B Ou, X Li, Y Zhao, R Ni, Y Shi, Pairwise prediction-error expansion for efficient reversible data hiding. IEEE Trans. Image Process. **22**(12), 5010–5021 (2013)
9. X Li, W Zhang, X Gui, B Yang, Efficient reversible data hiding based on multiple histograms modification. IEEE Trans. Inf. Forensics Secur. **10**(9), 2016–2027 (2015)
10. X Li, J Li, B Li, B Yang, High-fidelity reversible data hiding scheme based on pixel-value-ordering and prediction-error expansion. Signal Process. **93**, 198–205 (2013)
11. S Weng, J Pan, L Li, Reversible data hiding based on an adaptive pixel-embedding strategy and two-layer embedding. Inf. Sci. **369**, 144–159 (2016)
12. X Zhang, Reversible data hiding in encrypted image. IEEE Signal Processing Lett **18**(4), 255–258 (2011)
13. K Ma, W Zhang, X Zhao, N Yu, F Li, Reversible data hiding in encrypted images by reserving room before encryption. IEEE Trans. Inf. Forensics Secur. **8**(3), 553–562 (2013)
14. X Cao, L Du, X Wei, D Meng, X Guo, High capacity reversible data hiding in encrypted images by patch-level sparse representation. IEEE Trans Cybern **46**(5), 1132–1143 (2016)
15. A Nikolaidis, Reversible data hiding in JPEG images utilising zero quantised coefficients. IET Image Process. **9**(7), 560–568 (2015)
16. F Huang, X Qu, HJ Kim, J Huang, Reversible data hiding in JPEG images. IEEE Trans. Circuits Syst. Video Technol. **26**(9), 1610–1621 (2016)
17. B Ou, X Li, Y Zhao, R Ni, Efficient color image reversible data hiding based on channel-dependent payload partition and adaptive embedding. Signal Process. **108**, 642–657 (2015)
18. H Yao, C Qin, Z Tang, Y Tian, Guided filtering based color image reversible data hiding. J. Vis. Commun. Image Represent. **43**, 152–163 (2017)
19. C.C. Lo, .YC. Hu, A novel reversible image authentication scheme for digital images. Signal Process. 98, 174–185 (2014)
20. W Hong, M Chen, TS Chen, An efficient reversible image authentication method using improved PVO and LSB substitution techniques. Signal Process. Image Commun. **58**, 111–122 (2017)
21. H Wu, J Dugelay, Y Shi, Reversible image data hiding with contrast enhancement. IEEE Signal Processing Lett **22**(1), 81–85 (2015)
22. H Chen, J Ni, W Hong, T Chen, Reversible data hiding with contrast enhancement using adaptive histogram shifting and pixel value ordering. Signal Process. Image Commun. **46**, 1–16 (2016)
23. CC Chang, TD Kieu, YC Chou, in *TENCON 2007—2007 IEEE Region 10 Conference*. Reversible data hiding scheme using two steganographic images (IEEE, 2007), pp. 1–4
24. CC Chang, TC Lu, G Horng, YH Huang, YM Hsu, in *9th International Conference on Information, Communications and Signal Processing (ICICS 2013)*. A high payload data embedding scheme using dual stego-images with reversibility (IEEE, 2013), pp. 1–5
25. C Qin, CC Chang, TJ Hsu, Reversible data hiding scheme based on exploiting modification direction with two steganographic images. Multimed Tools Appl **74**(15), 5861–5872 (2015)
26. X Zhang, S Wang, Efficient steganographic embedding by exploiting modification direction. IEEE Commun. Lett. **10**(11), 781–783 (2006)
27. TC Lu, CY Tseng, JH Wu, Dual imaging-based reversible hiding technique using LSB matching. Signal Process. **108**, 77–89 (2015)
28. IF Jafar, KA Darabkh, RT Al-Zubi, RR Saifan, An efficient reversible data hiding algorithm using two steganographic images. Signal Process. **128**, 98–109 (2016)
29. CF Lee, KH Wang, CC Chang, YL Huang, in *Proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication*. A reversible data hiding scheme based on dual steganographic images (IEEE, 2009), pp. 228–237
30. CF Lee, YL Huang, Reversible data hiding scheme based on dual stegano-images using orientation combinations. Telecommun. Syst. **52**(4), 2237–2247 (2013)
31. TC Lu, JH Wu, CC Huang, Dual-image-based reversible data hiding method using center folding strategy. Signal Process. **115**, 195–213 (2015)
32. H Yao, C Qin, Z Tang, Y Tian, Improved dual-image reversible data hiding method using the selection strategy of shiftable pixels' coordinates with minimum distortion. Signal Process. **135**, 26–35 (2017)
33. J Fridrich, G Miroslav, R Du, Detecting LSB steganography in color, and gray-scale images. IEEE Multimed **8**(4), 22–28 (2001)