Multimedia Tools and Applications (2023) 82:6013–6032
https://doi.org/10.1007/s11042-022-13406-w

Reversible data hiding in encrypted images using median prediction and bit plane cycling-XOR

Fengyong Li[1,2] · Hengjie Zhu[1] · Chuan Qin[3]

Received: 17 November 2020 / Revised: 18 April 2022 / Accepted: 2 July 2022 /
Published online: 3 August 2022
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract
Existing reversible data hiding work in encrypted images (RDH-EI) mostly does not attain a good balance among good visual quality, large embedding capacity and high security performance. To address this problem, we design a new reversible data hiding scheme in encrypted images by combining median prediction and bit plan cycling-XOR. Our scheme firstly estimates the most significant bit (MSB) of each pixel by considering the median value of its adjacent pixels and generates a prediction error map to mark these pixels whose MSB bits are predicted incorrectly. Subsequently, we divide bit planes of cover image and then implement plane cyclic exclusive OR from least significant bit (LSB) plane to MSB plane. The LSB plane is finally vacated to be free room. Furthermore, the processed image is encrypted by a stream cipher algorithm, and data hider can embed additional data into the LSB plane. Separable operations of data extraction, image decryption and image recovery can be achieved successfully by the receiver. Comprehensive experiments demonstrate that compared with existing methods, our scheme can attain a better balance among good visual quality, large embedding capacity and high security performance.

Keywords Reversible data hiding · Image encryption · Median prediction · Bit plane division

1 Introduction

Data hiding technology usually hides additional secret data in multimedia files, such as digital files, digital images and digital audio or video, to achieve confidential communication [8, 28]. Digital images, as the popular multimedia file, are always used to the carriers (called as

✉ Fengyong Li
fyli@shiep.edu.cn

1 College of Computer Science and Technology, Shanghai University of Electric Power, Shanghai, 201306, People's Republic of China

2 Guangxi Key Lab of Multi-source Information Mining and Security, Guangxi Normal University, Guilin 541004, People's Republic of China

3 School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai, 200093, People's Republic of China

cover) to deliver the secret data. For example, during data hiding, secret data are embedded into digital images by modifying the elements of images (called as *marked image*), such as DCT coefficients or pixels. Correspondingly, secret data can be correctly extracted if the key is obtained by the recipient. However, an obvious pitfall is that the digital image may have a permanent modification during data embedding procedure. In this case, media content, e.g., DCT coefficients or pixels of the cover image, may be distorted due to data embedding, and cannot be recovered to original content after data extraction. In some real-world scenario, some images containing users' private information may be uploaded to social network platform. In order to ensure the images not being tampered and delivered illegally, the platform may embed the user's ID and authentication label to efficiently identify these images' ownership. However, data embedding procedure will inevitably make irreversible modification for image content. This is unacceptable in some application scenarios that are extremely sensitive to image content, such as military communications, medical images, legal forensics, and engineering drawings.

In order to meet this requirement, some researchers proposed reversible data hiding (RDH) technology to give a reliable solution, such as lossless compression based RDH schemes [7, 21, 26], difference extension based RDH schemes [15, 20] and histogram translation based RDH schemes [6, 16, 24]. In these schemes, data extraction can be not only achieved correctly, but the original image can be also recovered exactly.

One might think that traditional data hiding techniques, such as digital steganography and digital watermarking, also achieve both correct data extraction and perfect cover image recovery [10]. Nevertheless, RDH differs significantly from watermarking and steganography due to the special application context. Actually, digital watermarking is mainly to extract data robustly from a media file whose quality has deteriorated, while digital steganography embeds data into media file by an undetectable manner and accordingly it mainly conceals data perceptibility and hiding action [18]. Neither digital watermarking nor steganography cares about the exact recovery of the cover image.

Apparently, RDH methods can easily employ the pixel correlation to achieve data embedding. However, most of the existing RDH schemes are oriented to plaintext images, they can make good use of the image redundancy and spatial correlation of pixels. When the original images are encrypted by image encryption algorithm, the correlation between pixels may be greatly reduced or even disappeared [17]. In this scene, the application context poses extra requirements that existing plaintext image based RDH techniques do not support. First, the disturb of encryption procedure makes that the decryption image containing additional messages is difficult to obtain a good visual quality [2]. Second, it is well known that the correlation between pixels in encrypted images may disappear, it leads to RDH schemes in encrypted image cannot achieve a large embedding capacity [12]. Overall, encrypted image-based RDH needs to meet the requirements of (1) good visual quality of the marked image, (2) large embedding capacity for additional data, and (3) exact recovery for original image. Meeting these three requirements simultaneously and achieving a better balance among them is still an open challenge.

Facing the aforementioned requirements, we propose an effective encrypted image-based RDH solution. In general, the following novel contributions have been achieved:

– We propose a new encrypted image-based RDH method (RDH-EI) by designing median prediction method and bit plan division mechanism. Proposed scheme can not only extract additional data correctly from encrypted image, but the original image can be also recovered exactly. Our scheme can attain a better balance among high visual quality, large embedding capacity and high security performance.

– Our scheme can work over uncompressed natural images and can achieve separable operations of data extraction, direct image decryption, and lossless image recovery. We estimate the most significant bit (MSB) of each pixel by considering the median value of its adjacent pixels and then construct a prediction error map to mark these pixels whose MSB bits are predicted incorrectly. Subsequently, we divide multiple bit planes and perform plane cycle exclusive OR. Since plane cycle exclusive OR procedure can successfully transfer the predicted bit space to the LSB plane without affecting the bits in the high-bit plane, proposed scheme can thus obtain a large embedding capacity with keeping high visual quality.
– We implement comprehensive experiments over multiple classical testing images and a large-scale natural database, BOSSbase v1.01. Experimental results finally demonstrate that proposed scheme can attain a better balance in terms of good visual quality, large embedding capacity and high security performance.

The rest of the paper is organized as follows. Section 2 introduces a series of existing encrypted image-based RDH works. Section 3 provides the details of the proposed scheme. Subsequently, comprehensive experiments are performed to evaluate the performance of proposed scheme. The experimental results and corresponding discussions are presented in Section 4. Section 5 finally concludes this paper.

2 Related work

In general, existing RDH-EI schemes are mainly divided into two categories: vacate room after encryption (VRAE) [5, 23, 29–31] and vacate room before encryption (VRBE) [3, 9, 11, 14, 27]. We briefly introduce existing related works according to these two categories.

Firstly, for VRAE category, the original image is firstly encrypted by the content owner. Data hider designs specific algorithm to vacate free rooms in the encrypted image and then embeds additional data and auxiliary data into free rooms. Zhang et al. [30] employed data compression mechanism to design a separable reversible data hiding scheme. Encrypted images were firstly grouped and the low bits of each group of pixels were compressed to vacate free rooms. However, due to a significant modifications in low bits of image and low compression efficiency, the embedding capacity of Zhang's scheme is relatively small. On the basis, Zhang et al. [29] further designed improved solution by using bit substitution, in which the low bits of encrypted image were replaced directly with the bits of additional data. Meanwhile, the embedded bits and the pseudo-random bit sequence generated by the modulation of additional bits were combined to replace the remaining low-bit bits. Although this scheme improves embedding capacity, the additional data may contain error bits when they are extracted by the receiver. Wu et al. [23] replaced the MSB of encrypted pixels with the additional data. Unfortunately, modifying the MSB of encrypted images resulted in a poor quality of decrypted images. In order to further improve the performance, Zheng et al. [31] used hamming distance to design new RDH scheme. By calculating the Hamming distance between LSB bits and auxiliary bits, the LSBs of pixels in encrypted image were lossless compressed to vacate room for additional data. Huang et al. [5] combined the stream cipher and prediction error to vacate room for data embedding. The permutation operation was performed over encrypted image to improve the security of cipher stream. However, for the split natural image or the image with little pixel similarity, the embedding capacity has no significant improvement. Overall, in VRAE framework, since encryption procedure destroys the spatial correlation of the original image, it is difficult to realize large-capacity

additional data embedding through image compression. In other words, in VRAE framework, RDH schemes have to sacrifice the security of encryption algorithm to meet high embedding capacity requirement.

Secondly, regarding VRBE category, VRBE framework usually creates embedding room in the plaintext domain, and then encrypts the processed image before delivering it to data hider. In essential, the content owner is expected to perform an extra preprocessing before encryption in VRBE framework. In this context, Nguyen et al. [14] proposed a reversible data hiding method in encrypted images by pixel division. In this scheme, half of pixels were used to classify the rest of the pixels into smooth and complex regions to provide room for embedding additional data. This additional embedded data can be extracted exactly and the encrypted image can be reconstructed precisely to its original version. Nevertheless, since half pixels of cover image belong to the smooth region and they must keep unchanged in the whole process, the maximum of embedding rate is only 0.5 bpp with the parameter n=1. Li et al. [9] employed the prediction error to segment images into 3×3 blocks, and then used the pixels at four corners to predict the remaining five pixels. With a given threshold, the secret data were embedded into the locations where the predicted pixels can meet the conditions in each block. Chen et al. [3] used linear regression-based predictor to improve the accuracy of predictions, and a prediction error map is constructed to eliminate errors caused by inaccurate predictions. Although Chen's scheme can achieve the reversibility without error for the original image, the auxiliary data lower the embedding capacity and increases the time complexity of embedding/extracting data and recovering images. Yin et al. [27] proposed a high-capacity RDH-EI algorithm based on multi-MSB prediction and Huffman coding. Multi-MSB of each pixel is firstly predicted by median prediction and marked by Huffman coding in the original image. Then, the image was encrypted by a stream cipher method and the vacated room can be finally used to embed additional data by multi-MSB substitution. However, since the multi-MSBs are modified in data embedding procedure, the visual quality of the decrypted image containing additional data has a significant degradation. Li et al. [11] further designed double linear regression prediction model to construct new reversible data hiding scheme in encrypted images, which can significantly improve the prediction accuracy of current pixel based on neighboring pixels. Nevertheless, affected by image content, double linear regression prediction model may produce more prediction error in smooth images with respect to the texture images.

According to the above detailed analysis, VRAE framework based RDH schemes always vacate the free rooms after encryption procedure to embed additional data, they can thus only obtain a low embedding capacity. While for VRBE framework based RDH schemes, although many works have been developed to gradually improve the data hiding performance, these works either sacrifice embedding capacity to obtain good visual quality, or sacrifice visual quality to gain high embedding capacity. To the best of our knowledge, few schemes have achieved a satisfactory trade-off among better visual quality, larger embedding capacity and higher security performance. This paper tries to provide a solution in this regard.

3 Proposed scheme

3.1 The framework of proposed scheme

Proposed scheme contains three parties: content owner, data hider and receiver. Firstly, the content owner predicts the most significant bit (MSB) of each pixel by considering the

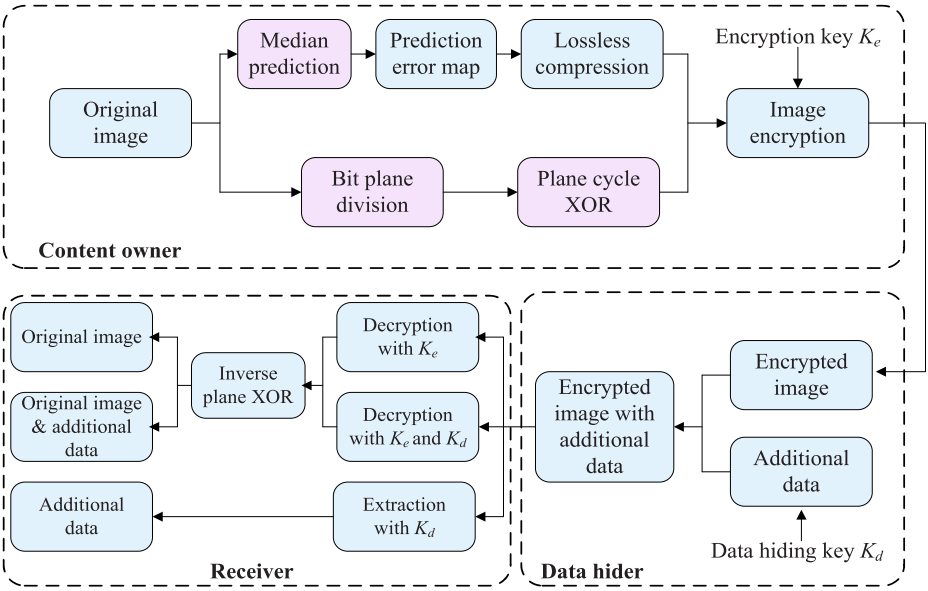


Fig. 1 The framework of proposed scheme

median value of its adjacent pixels. The corresponding prediction error map is constructed to mark the pixels whose MSB bits are predicted incorrectly. Furthermore, eight bit planes of image are divided and then implemented plane cycling-XOR to vacate LSB plane. The processed image is subsequently encrypted by encryption key and delivered to the data hider. Secondly, when the data hider receives the encrypted image from the content owner, secret data are embedded into the encrypted image by the hiding key. Finally, when the receiver obtains the encrypted image, he can conduct separable operations of data extraction, image decryption and image recovery according to the keys he has. The overall framework of proposed scheme is shown in Fig. 1.

3.2 Median prediction and prediction error map

Since the neighboring pixels in natural image usually maintain strong correlation, the prediction model can be built by using the consistency relationship between the neighboring pixels. For a natural image, since the neighborhood correlation of MSB plane is higher than that of other bit planes, the prediction accuracy of MSB plane is generally higher. Considering the above analysis, we construct a median prediction model that works over the MSB plane.

Assume that $x_{i,j}$ is the pixel that will be predicted in the original image $\mathbf{X}$ with size $M \times N$, $1 \leq i \leq M$, $1 \leq j \leq N$. We construct 2×2 pixel block to build median prediction model. Figure 2 shows four different construction methods of 2×2 pixel blocks, where $x_{i,j}$ is the current pixel in each construction mode. Notably, since the closer distance between neighborhood pixels implies a stronger correlation, smaller pixel blocks can effectively avoid inaccurate pixel prediction caused by sudden changes in edge or texture regions, and can greatly improve the prediction accuracy of the highest bit plane.

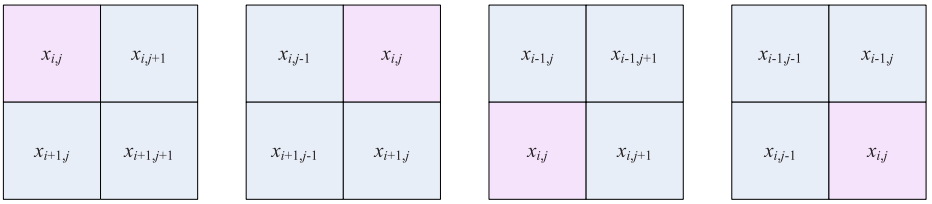


Fig. 2 Four different construction methods for 2×2 pixel blocks, where $x_{i,j}$ is the current pixel

Without loss of generality, we choose Fig. 2(d) to describe the algorithm in this paper. The prediction value $x'_{i,j}$ can be calculated by the following equation.

$$x'_{i,j} = \begin{cases} \min\left(x_{i-1,j}, x_{i,j-1}\right) & , \text{ if } x_{i-1,j-1} \leq \min\left(x_{i-1,j}, x_{i,j-1}\right) \\ \max\left(x_{i-1,j}, x_{i,j-1}\right) & , \text{ if } x_{i-1,j-1} \geq \max\left(x_{i-1,j}, x_{i,j-1}\right) \\ x_{i,j-1} + x_{i-1,j} - x_{i-1,j-1} & , \text{ Otherwise} \end{cases} \tag{1}$$

Since the neighborhood pixels have a stronger correlation, median predictor can effectively collect their correlation characteristics and perform an efficient prediction, resulting in a significant improvement of prediction accuracy for MSB plane. Furthermore, the difference $e_{i,j}$ between the original pixel and its predicted value is calculated by using (2).

$$e_{i,j} = x_{i,j} - x'_{i,j} \tag{2}$$

Subsequently, we calculate the prediction error value $t_{i,j}$ according to the (3) and consider $t_{i,j}$ as a mark to record whether the MSB of current pixel $x_{i,j}$ is the same as the MSB of its predicted value. Finally, a prediction error map is generated by collecting all marks, whose size is close to that of the original image.

$$t_{i,j} = \begin{cases} 0 & , \text{ if } e_{i,j} \in [-127, 127] \\ 1 & , \text{ Otherwise} \end{cases} \tag{3}$$

The prediction error map indicates whether the MSB of each original pixel is accurately predicted. In order to achieve perfect recovery for original image, we consider the prediction error map as auxiliary data and deliver them along with the additional data. Obviously, if the auxiliary data are too much, they will inevitably take up more embedding room, resulting in greatly reduction for additional data.

Fortunately, since our designed median predictor is very accurate in predicting the MSB plane, the prediction error map shows a relatively uniform phenomenon, that is, most of the values in this map are 0. This implies that we can employ lossless compression method, e.g., run-length coding compression [4], to further compress the auxiliary data.

3.3 Bit plane cycling-XOR

When the prediction error map is calculated, the content owner divides eight bit planes of original image, and then performs plane cycling-XOR process. Denote that $x_{i,j}$ and $\bar{x}_{i,j}$ are original pixel and the pixel after cycling XOR, respectively, and $\{x^8_{i,j}x^7_{i,j}\cdots x^k_{i,j}\cdots x^1_{i,j}\}$ and $\{\bar{x}^8_{i,j}\bar{x}^7_{i,j}\cdots \bar{x}^k_{i,j}\cdots \bar{x}^1_{i,j}\}$ are their corresponding binary representations, where $x^k_{i,j}$ stands for the bit value of the k-th plane and $x^8_{i,j}$ is the MSB bit. Bit plane cycling-XOR process XORs the current bit plane and its adjacent bit plane, and is sequentially performed from

the 8-th bit plane to LSB bit plane. Figure 3 shows an actual procedure of bit plane cycling-XOR.

$$\bar{x}_{i,j}^{k} = \begin{cases} x_{i,j}^{k} & , \ k = 1 \\ x_{i,j}^{k} \oplus x_{i,j}^{k-1} & , \ k = 2, 3, \cdots, 8 \end{cases} \tag{4}$$

According to (4), the XOR processing is implemented between the $(k-1)$-th bit plane and the k bit plane to generate a new bit plane, which is used to replace the original k-th bit plane, that is to say, the information of each bit plane is stored lossless in the upper bit plane. Accordingly, the LSB bit plane can be vacated as the free rooms, which are used to embed secret data (including auxiliary data and additional data).

3.4 Image encryption

Furthermore, the content owner embeds the auxiliary data (including compressed prediction error map and its position information) into the vacated room (LSB plane). Subsequently, the stream cipher is used to encrypt the processed image.

Denote the processed image as $\overline{\mathbf{X}} = \{\bar{x}_{i,j}\}$ and the binary representation of each pixel as $\bar{x}_{i,j} = \{\bar{x}_{i,j}^{8}\bar{x}_{i,j}^{7} \cdots \bar{x}_{i,j}^{k} \cdots \bar{x}_{i,j}^{1}\}$, where (i, j) stands for the pixel position. We can describe the image encryption procedure by the following steps.

Step 1 Calculate the binary representation of each pixel as follows.

$$\bar{x}_{i,j}^{k} = \left\lfloor \frac{\bar{x}_{i,j}}{2^{k-1}} \right\rfloor \bmod 2, \qquad k = 8, 7, 6, \cdots, 1 \tag{5}$$

Step 2 Use the encryption key K_e to generate a standard random bit sequence $\mathbf{R} = \{r_{i,j}\}$ with length of $M \times N \times 8$.

Step 3 Bit sequence $\mathbf{R}$ is used to encrypt the processed image by the following equation.

$$\hat{x}_{i,j}^{k} = \bar{x}_{i,j}^{k} \oplus r_{i,j}^{k}, \qquad k = 8, 7, 6, \cdots, 1 \tag{6}$$

where $\hat{x}_{i,j}^{k}$ is the encrypted version of corresponding bit $\bar{x}_{i,j}^{k}$ in encrypted image.

Notably, since the stream cipher only changes the pixel value, the position of each pixel in the image will not be changed after encryption. Finally, the content owner delivers the encrypted image to the data hider.

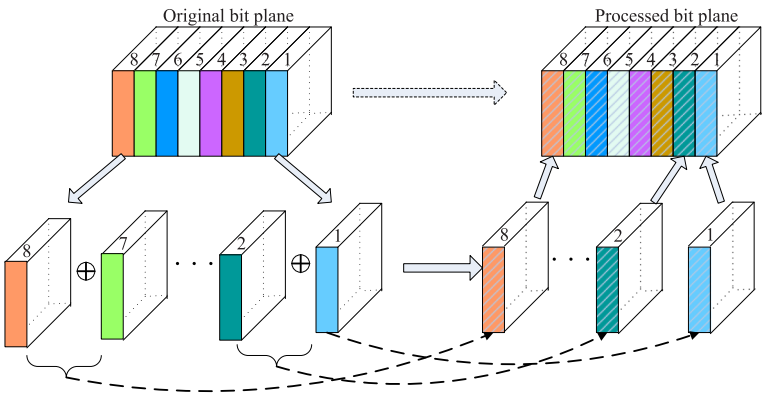


Fig. 3 The procedure of bit plane cycling-XOR

3.5 Data embedding

Since the auxiliary data and additional data are both embedded into the LSB plane, the theoretical maximum embedding capacity for additional data is $M \times N$ bits. Nevertheless, we should note that different images have different spatial correlation, leading to a different prediction error map. Therefore, the actual embedding capacity for additional data is also different. Assume that the length of compressed prediction error map is L_t bits, and the length of its position information is L_a bits, the actual maximum embedding capacity for additional data can be calculated as follows.

$$L_c = M \times N - L_t - L_a \tag{7}$$

When the data hider receives the encrypted image, he can easily obtain the auxiliary data and determine the starting position information of compressed prediction error map. Based on the auxiliary data, the detailed data embedding procedure can be described as follows.

Step 1 Extract the auxiliary data and then determine the starting position information of compressed prediction error map.

Step 2 Compress the prediction error map and calculate its length as L_e.

Step 3 Calculate the length of additional data by (7), and then attach the additional data after auxiliary data as the complete secret data.

Step 4 With the data hiding key K_d, the secret data can be embedded sequentially by directly replacing the LSBs of pixels in processed image.

After the additional data are embedded, encrypted image containing the secret data is delivered to the receiver through the public network channel. Notably, since proposed median prediction model can obtain a higher prediction accuracy, the prediction error map can be losslessly compressed to a small storage space, making a higher embedding capacity for additional data.

3.6 Data extraction and image recovery

For the receiver, when the encrypted image containing secret data is obtained, he can achieve the separable operations of data extraction, image decryption and image recovery according to the obtained key: (1) only the image decryption key K_e, (2) only the data hiding key K_d, and (3) both the image decryption key K_e and the data hiding key K_d.

– Only the image decryption key K_e.

  Step 1: The key K_e is used to generate random binary bit sequence $\mathbf{R} = \{r_{i,j}\}$, which is used to decrypted the image as $\overline{\mathbf{X}} = \{\bar{x}_{i,j}\}$.

  Step 2: Extract auxiliary data from the LSB plane of decrypted image and decompress the prediction error map.

  Step 3: According to (1), calculate the predicted value of each pixel and denote it as $\mathbf{X}''_{i,j} = \{x''_{i,j}\}$.

  Step 4: Combining with the prediction error map $\mathbf{T}_{i,j} = \{t_{i,j}\}$ and $\mathbf{X}''_{i,j} = \{x''_{i,j}\}$, calculate the MSB plane of original image as follows.

$$b_{i,j} = \left\lfloor \frac{x''_{i,j}}{2^8} \right\rfloor \tag{8}$$

$$\tilde{x}_{i,j}^{8} = \begin{cases} \overline{b_{i,j}} & , \text{ if } t_{i,j} = 1 \\ b_{i,j} & , \text{ if } t_{i,j} = 0 \end{cases} \tag{9}$$

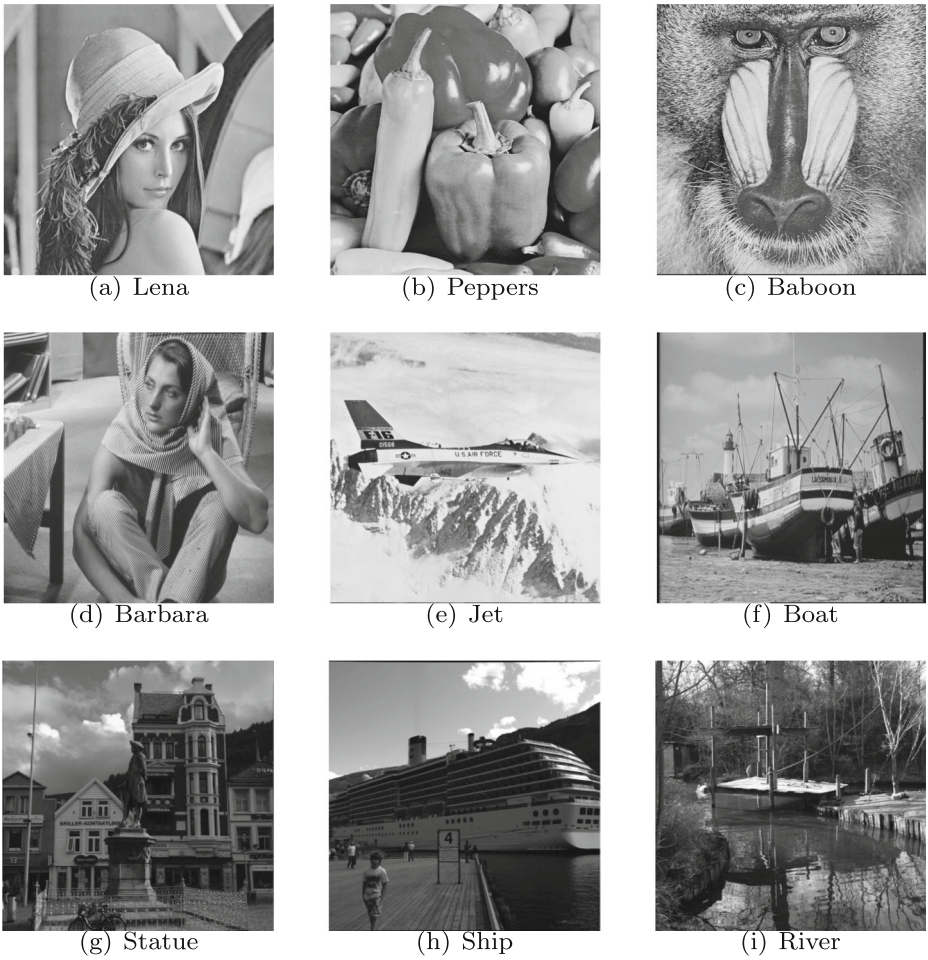
where $\overline{b_{i,j}}$ is the flipped bit of $b_{i,j}$ and $\tilde{x}_{i,j}^{8}$ is the recovered MSB bit for original pixel $x_{i,j}$.

Step 5: Implement inverse cyclic XOR processing from the MSB plane to LSB plane according to the following equation.

$$x_{i,j}^{k} = \begin{cases} \tilde{x}_{i,j}^{k} & , \ k = 8 \\ x_{i,j}^{k+1} \oplus \tilde{x}_{i,j}^{k+1} & , \ k = 7, 6, \cdots, 1 \end{cases} \tag{10}$$

– **Only the data hiding key** K_d.

The secret data are extracted from LSB plane by using the data hiding key and the additional data are then separated from the auxiliary data.



(a) Lena (b) Peppers (c) Baboon

(d) Barbara (e) Jet (f) Boat

(g) Statue (h) Ship (i) River

Fig. 4 Two experimental image sets, classical testing set and BOSSbase v1.01, are used in our experiments, where the images (a)-(f) are from the classical testing set and the images (g)-(i) are from BOSSbase v1.01

– **Both encryption key and data hiding key**.
If the receiver obtains both the encryption key and the data hiding key, the additional data can be extracted correctly and the original image can be also recovered lossless.

4 Experimental results and analysis

In this section, we evaluate the proposed scheme by a series of experiments. Two different image sets are used, one is the classical gray testing image set, including Lena, Peppers, Baboon, Barbara, Jet and Boat, as shown in Fig. 4, and another is the large-scale natural image database BOSSbase v1.01 [1]. All experimental images are resized with 512×512 and the bit per pixel (*bpp* for short) is considered as the measurement of embedding rate, which represents the average number of bits carried by each pixel.

In order to give sufficient comparisons, we introduce two standard measurements, peak signal to noise ratio (PSNR) and structural similarity index measurement (SSIM), to evaluate the performance of different methods, where PSNR presents the peak signal to noise ratio and SSIM mainly presents the structural similarity between the original image and the decrypted image containing secret data. Their calculations are shown as follows.

$$MSE = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} ||\mathbf{X}(i, j) - \mathbf{I}(i, j)||^2 \tag{11}$$

$$\text{PSNR} = 10 \cdot \log_{10} \left(\frac{MAX_{\mathbf{I}}^2}{MSE} \right) \text{dB} \tag{12}$$

where $\mathbf{X}$ represents the original image and $\mathbf{I}$ is the decrypted image containing secret data. MSE is the mean square error and $MAX_{\mathbf{I}}$ equals to 255 for gray image. In general, a larger PSNR implies that the decrypted images have a higher visual quality.

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2\mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{13}$$

where μ_x and μ_y are the average values of $\mathbf{X}$ and $\mathbf{I}$, respectively, and σ_x and σ_y are the standard deviation values. σ_{xy} is the covariance of $\mathbf{X}$ and $\mathbf{I}$ and c_1 and c_2 are constants. Apparently, SSIM value is between 0 and 1, and the closer the value is to 1, the more similar $\mathbf{X}$ and $\mathbf{I}$ are.

4.1 Effectiveness testing for proposed scheme

In order to evaluate the effectiveness of proposed algorithm, we carry a series of experiments over three classical testing images, Lena, Baboon and Peppers, where Baboon is a texture image and Lena and Peppers stand for the smooth images. The corresponding experimental results are shown in Fig. 5. It can observe that the original image content is significantly different from their encryption version, e.g., Figs. 5(b), 5(g) and 5(l). Moreover, when the secret data are embedded into the encrypted images, e.g., Figs. 5(c), 5(h) and 5(m), the difference between the encrypted images and the marked images (the encrypted images containing secret data) is very hard to be intuitively noticed. This demonstrates that the encryption algorithm of proposed scheme has a good scrambling effect.

In addition, we find that for different experimental images, Lena, Baboon and Peppers, the maximum embedding rates can approximately get 0.9960 bpp, 0.9960 bpp and 0.9961

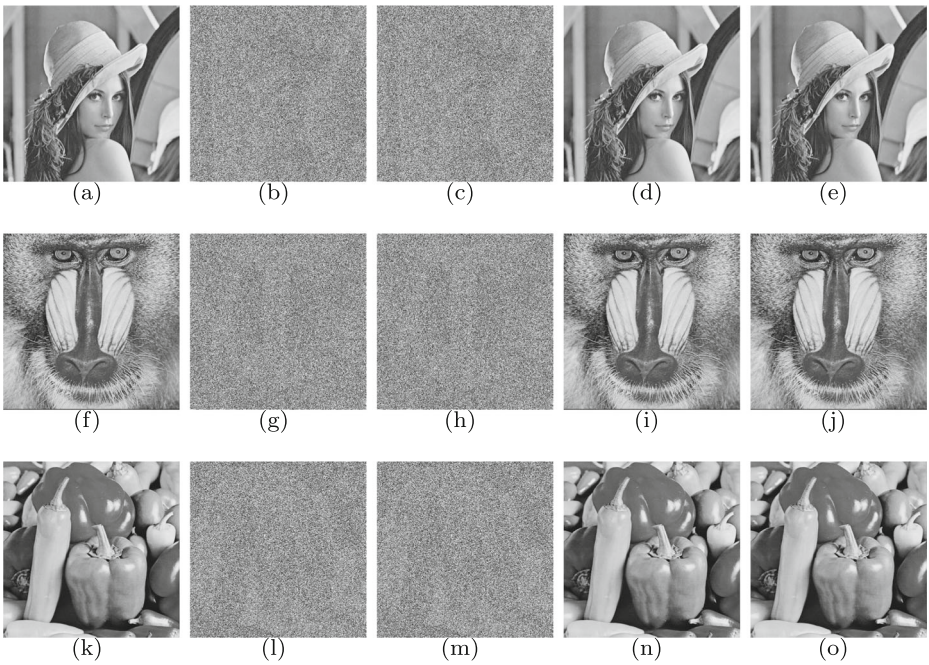


Fig. 5 Overall effectiveness testing for proposed scheme. Three classical testing images, Lena, Peppers and Baboon, are used in this experiment. The first column shows the original images, the second column is the encrypted images, the third column is the encrypted images containing secret data, the forth column is the decrypted images containing secret data and the fifth column shows the recovered images

bpp, respectively. Correspondingly, their decrypted versions coincidentally show significant high PSNR values, e.g., 51.1713 dB for Fig. 5(d), 51.1575 dB for Fig. 5(i) and 51.1599 dB for Fig. 5(n). These results imply that high visual quality can be obtained whatever texture or smooth images are used. Actually, this is mainly because the MSB bits of adjacent pixels always keep consistent with a greater probability, proposed median predictor can thus perform an accurate prediction for MSB plane. Accordingly, the prediction errors are usually very rare and relatively concentrated so that prediction error map can be compressed with a large compression ratio, resulting in a relatively large vacated room. Finally, the recovered images can obtain PSNR $= \infty$, e.g., Figs. 5(e), 5(j) and 5(o). This further verifies that by combining median prediction model and bit plane cycling-XOR, proposed scheme can achieve data extraction correctly and perfect recovery of original image.

4.2 Performance comparison for embedding capacity

In our scheme, we only consider LSB plane to embed the secret data and thus the maximum embedding rooms should be $M \times N$. According to (7), the length of compressed prediction error map is a major influence factor for the embedding capacity of additional data. The less prediction errors for MSB bits, the smaller the compressed prediction error map and the higher embedding capacity of additional data.

To show the advantage of proposed prediction model, we evaluate embedding capability of additional data by comparing two existing prediction models, Li's prediction model [11]

and Yin's prediction model [27]. A series of experiments are implemented to show the performance of proposed scheme. In order to give a fair comparisons, we define E_p to represent the bit number of correct prediction in different prediction models. Then,

$$E_p = M \times N - \sum_{i,j} e_{i,j} \tag{14}$$

where $\sum_{i,j} e_{i,j}$ is the number of prediction errors in the prediction error map. Apparently, a larger E_p means the higher compression ratio for prediction error map.

Table 1 shows the corresponding testing results. As can be seen from this table that proposed model has a significant higher E_p than that of Li's model and Yin's model over the classical image set. Correspondingly, a higher embedding rate (*ER* for short) can be easily obtained. In addition, we also test the average performance of different prediction models over the large-scale natural image set BOSSbase. Our model can get an average prediction accuracy closing to 99.61%. This means that the prediction error map of the proposed scheme is more suitable for compression, leading to a higher embedding rate for additional data.

Table 1 Performance comparison of prediction error E_p and embedding rate ER for three prediction models, Li's model [11], Yin's model [27] and proposed model

Source	Images	Prediction model	E_p	ER
Classical set	Lena	Li's model [11]	226932	0.7352
		Yin's model [27]	228520	0.7910
		Proposed	262144	0.9961
	Peppers	Li's model [11]	239070	0.8278
		Yin's model[27]	213781	0.4385
		Proposed	262144	0.9961
	Baboon	Li's model [11]	257915	0.9755
		Yin's model [27]	124904	0.2163
		Proposed	262136	0.9960
	Barbara	Li's model [11]	232891	0.7807
		Yin's model [27]	238567	0.6023
		Proposed	262144	0.9961
	Jet	Li's model [11]	204853	0.5698
		Yin's model [27]	216138	0.6006
		Proposed	262144	0.9961
	Boat	Li's model [11]	201963	0.5447
		Yin's model [27]	211781	0.6588
		Proposed	262144	0.9961
Natural set	BOSSbase	Li's model [11]	204230	0.7791
		Yin's model [27]	187530	0.8307
		Proposed	261110	0.9959

Two testing image sets, classical set and natural set (BOSSbase), are used in this experiment

Table 2 Average maximum embedding rate (AMER) for seven RDH schemes by using the natural image set BOSSbase

Scheme	Proposed	Zheng [31]	Chen [3]	Ma [12]	Li [9]	Li [11]	Malik [13]
AMER(bpp)	0.996	0.250	0.780	0.641	0.999	0.779	0.741

To gain more insight between our scheme and other schemes, we use the BOSSbase to test the average maximum embedding rate (AMER) of six existing schemes, Zheng's scheme [31], Ma's scheme [12], Li's scheme [9], Chen's scheme [3], Li's scheme [11], and

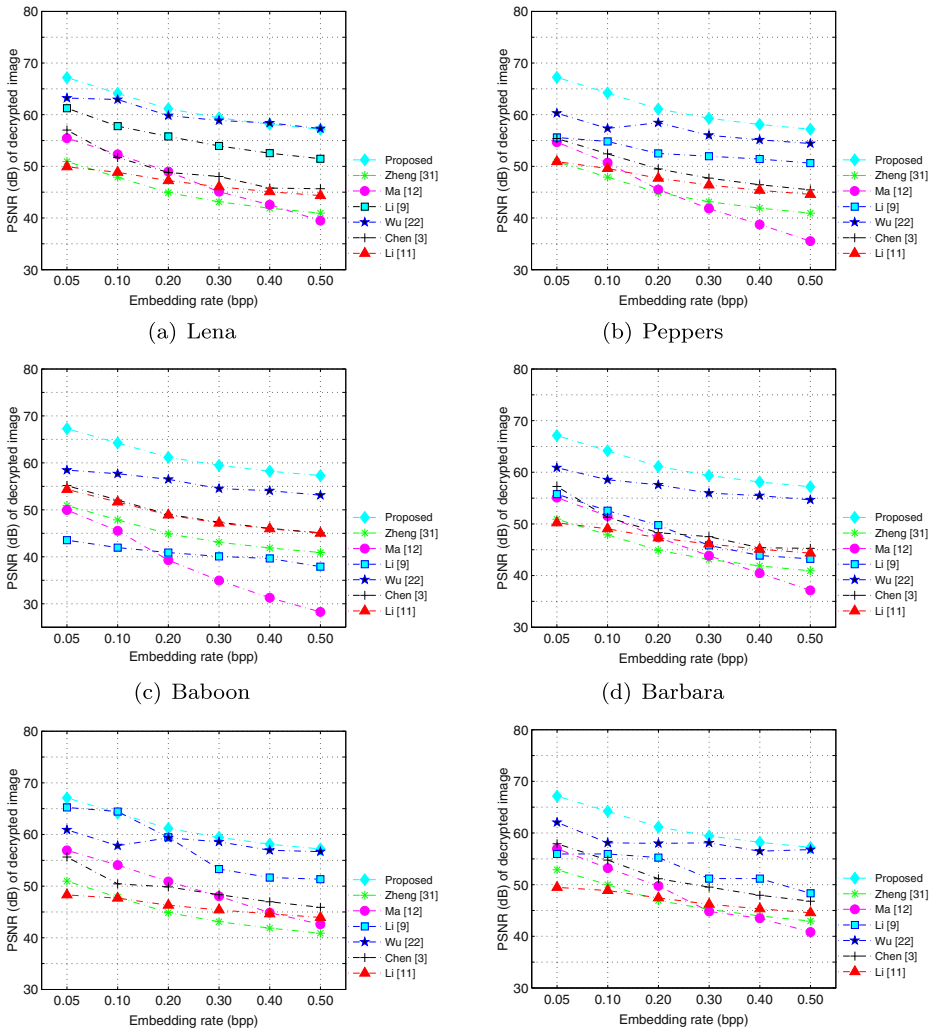


Fig. 6 Overall performance comparison for different schemes with six classical testing images, (a)Lena, (b)Peppers, (c)Baboon, (d)Barbara, (e)Jet and (f)Boat

Malik's scheme [13]. The actual testing results are shown in Table 2. As can be observed that the average maximum embedding rate of proposed scheme is significantly higher than that of other several schemes. We explain this interesting phenomenon as follows. Since proposed median prediction model can predict accurately the MSB plane of original image, the prediction error map forms a distribution that is approximately all 0 (corresponding to (3)) and the auxiliary data can therefore be further compressed with a smaller length, leading to embedding more additional data.

4.3 Performance comparison for visual quality

For RDH-EI schemes, the visual quality of decrypted images and recovered images is an important performance measurement. In order to show the advantages of the proposed scheme, we implement a series of experiments to compare proposed scheme with six existing RDH-EI schemes, Zheng's scheme [31], Ma's scheme [12], Wu's scheme [22], Li's scheme [9], Chen's scheme [3], and Li's scheme [11]. Each scheme is implemented over six classical testing images, Lena, Peppers, Baboon, Barbara, Jet and Boat.

To give a fair comparison, the PSNR values of decrypted image containing secret data are calculated as the testing results. The corresponding experimental results are shown in Fig. 6. In these figures, the abscissa represents the embedding rate, while the ordinate stands for the PSNR value of the decrypted image containing secret data. We test six different embedding rates, 0.05 bpp, 0.10 bpp, 0.20 bpp, 0.30 bpp, 0.40 bpp, 0.50 bpp, to provide a fair comparison. We can easily observe that with the same embedding rate, proposed scheme has a significant superior visual quality comparing with other existing schemes, no matter what experimental images are used. The average PSNR gain is more than 8 dB-10 dB. This is mainly because proposed median prediction model can predict well the MSB plane for original image so that the prediction error map can be compressed with a larger compression ratio. Accordingly, with the same additional data, smaller auxiliary data necessarily produce less LSB modifications for the decrypted image, resulting in a higher visual quality.

In addition, in order to further show the advantages of the proposed scheme, we also test the performance of different schemes with four larger embedding rates, 0.60 bpp, 0.70 bpp, 0.80 bpp, 0.90 bpp. Notably, since some schemes, e.g., Ma's scheme [12], Li's scheme [9], and Malik's scheme [13], are not completely suitable for large embedding rates, we do not

Table 3 Performance comparison in PSNR (dB) and SSIM for six existing schemes and proposed scheme. BOSSbase database is used to give the average testing results and four high embedding rates (bpp), 0.60, 0.70, 0.80, 0.90, are tested in this experiment

ER(bpp)	Measure	Scheme						
		Zheng [31]	Chen [3]	Ma [12]	Li [9]	Li [11]	Malik [13]	Proposed
0.60	PSNR	41.2162	44.8765	27.3720	40.5786	45.0322	51.1976	57.8711
	SSIM	0.9886	0.9923	0.8239	0.9947	0.9912	0.9854	0.996
0.70	PSNR	40.3691	44.2957	–	40.1804	44.2829	48.2644	56.2777
	SSIM	0.9856	0.9910	–	0.9937	0.9901	0.9712	0.9995
0.80	PSNR	39.3925	44.0166	–	–	43.9388	–	55.9764
	SSIM	0.9836	0.9894	–	–	0.9872	–	0.9994
0.90	PSNR	38.9799	43.3378	–	–	43.2681	–	55.1094
	SSIM	0.9834	0.9888	–	–	0.9865	–	0.9991

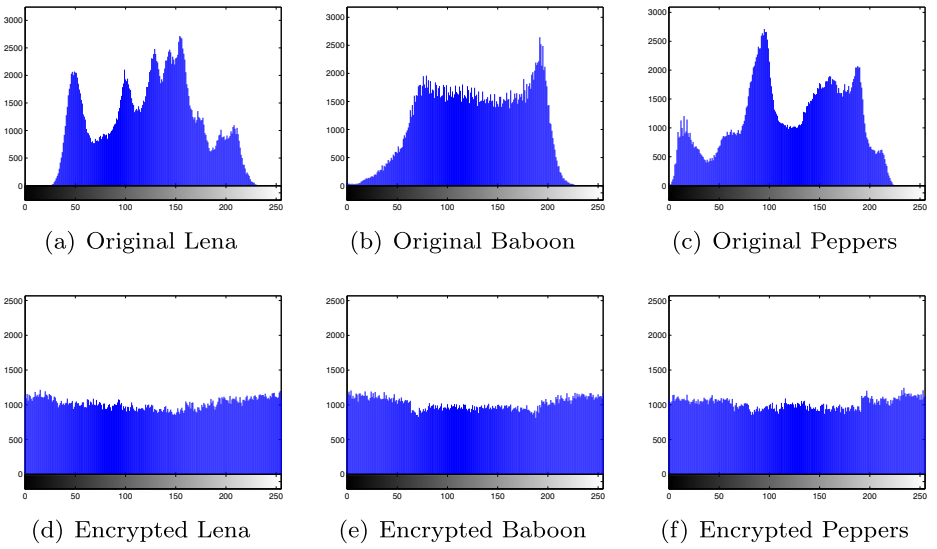
draw the corresponding curves of high embedding rate in Fig. 6. To ease understand, we take the classical BOSSbase database to test the average results. Table 3 shows the experimental results by comparing with six existing schemes, Zheng's scheme [31], Ma's scheme [12], Li's scheme [9], Chen's scheme [3], Li's scheme [11], and Malik's scheme [13]. As can be seen from this table, the proposed scheme consistently gets the highest PSNR value and SSIM value no matter which embedding rate is used. To be specific, when the embedding rate is 0.60 bpp, the PSNR value of proposed scheme is more than that of other methods with an average gain 15 dB, and the SSIM of proposed scheme can attain 0.9996. When the embedding rate is 0.90 bpp, the average gain of PSNR still reach 13 dB and the average SSIM can also get 0.9991. This demonstrates that proposed scheme always keeps a superior performance in visual quality comparing with existing state-of-the-art schemes, even if for higher embedding rates.

Notably, from Table 3 and Fig. 6, we can observe that proposed scheme shows a superior average performance than other existing several schemes. This phenomenon is easily explained as follows. Zheng's scheme [31] employs Hamming coding to perform lossless compression, but, the amount of compressed data is still much larger so that it seriously affects the number of embedded bits in each pixel, resulting in a lower embedding capacity. In addition, Chen's scheme [3] and Li's scheme [11] use single-layer linear regression and double-layer linear regression prediction models, respectively. Although these two methods have little difference in embedding rate and visual quality, their prediction accuracy still cannot be significantly improved due to a small number of mis-predictions. Wu's scheme [22] involves pixel block division to embed secret information into smooth pixel blocks. This scheme can achieve high visual quality, it, however, relies heavily on the number of smooth pixel blocks for natural image, and does not have stable robustness performance for a large-scale image database. Since neighborhood pixels have a stronger correlation in MSB plane, with designed median prediction model, proposed scheme can achieve a higher prediction accuracy for MSB plane, eventually approaching 100%. This makes proposed method only needs to embed less auxiliary data. Accordingly, the visual quality of decrypted image containing secret data can be significantly improved when the same amount of additional data are embedded.

4.4 Security analysis

First, we test the security of proposed encryption scheme by measuring histogram and pixel correlation. The gray histogram can reflect the distribution of the pixels in an image. If the grayscale distribution of the encrypted image is irregular, it is very difficult for the attacker to correctly recover the image by analyzing the pixel gray values. Accordingly, we test the histogram results before and after encryption for Lena, Peppers and Baboon, respectively, and then show the results in Fig. 7. Obviously, original histograms shows a specific patterns (diverse histogram distribution), but, the pixels are distributed uniformly after encryption. It implies that proposed encryption algorithm can obtain a good scrambling effect. Moreover, a good encryption scheme must effectively reduce the correlation between adjacent pixels. The correlation coefficient is further used to analyze the correlation among adjacent pixels and can be calculated as follows.

$$r_{xy} = \frac{\mathrm{cov}(x, y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{15}$$



(a) Original Lena (b) Original Baboon (c) Original Peppers

(d) Encrypted Lena (e) Encrypted Baboon (f) Encrypted Peppers

Fig. 7 The histogram analysis of original image and encrypted image for three classical images, Lena, Baboon, Peppers. The first row shows the histogram of original images, and the second row is the histogram of encrypted images

where

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{16}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))^2 \tag{17}$$

$$\mathrm{cov}(x, y) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))(y_i - E(y)) \tag{18}$$

In above equations, r_{xy} close to 1 implies that the adjacent pixels have a strong correlation and r_{xy} close to 0 means a weak correlation. Correspondingly, 2500 pairs of two-adjacent pixels in the horizontal, vertical and diagonal directions are randomly selected from the original images and its encrypted version, respectively, and then are used to calculate the correlation coefficients. The corresponding results are shown in Table 4. We can observe

Table 4 The correlation coefficients of original image and corresponding encrypted image

Image	Original image			Encrypted image		
	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Lena	0.9695	0.9832	0.9688	-0.0038	-0.0077	-0.0213
Peppers	0.9843	0.9856	0.9694	-0.0080	0.0077	-0.0161
Jet	0.9758	0.9709	0.9510	0.0055	-0.0087	-0.0197
Barbara	0.8524	0.9602	0.8325	-0.0126	-0.0058	-0.0046
Baboon	0.8777	0.7859	0.7465	-0.0052	0.0167	0.0039

that the correlation coefficients of the original images are close to 1, while the correlation coefficients of the ciphered images approach 0. The results demonstrate that our encryption algorithm can effectively reduce the correlation of adjacent pixels.

Second, in order to show the security of the proposed scheme, we further test the entropy of encrypted image and key sensitivity. Image entropy is used to measure the average amount of information in the image and its maximum theoretical value is 8 [19].

$$H(s) = -\sum_{i=0}^{2^n-1} p(s_i) \cdot \log_2(p(s_i)) \tag{19}$$

where s represents the information source, $p(s_i)$ represents the probability of the symbol s_i, 2^n represents the total states number of the information source, and is generally set to 256 for grayscale images. Moreover, key sensitivity can be measured by the Unified Average Changing Intensity (UACI) and the Number of Pixels Change Rate (NPCR).

$$UACI = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{|c_1(i,j) - c_2(i,j)|}{255} \times 100\% \tag{20}$$

$$NPCR = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} D(i,j) \times 100\% \tag{21}$$

$$D(i,j) = \begin{cases} 0, & p(i,j) = e(i,j) \\ 1, & \text{otherwise} \end{cases} \tag{22}$$

Table 5 Security performance comparison for different schemes over Lena and Baboon. Three security measurements, entropy, UACI, and NPCR, are tested in this experiment

Schemes	Images	Entropy			UACI(%)	NPCR(%)
		Original	Encrypted	Difference		
Proposed	Lena	7.4456	7.9980	0.5524	48.4467	99.9997
	Baboon	7.4745	7.9984	0.5239	50.3374	99.9996
Chen [3]	Lena	7.4456	7.9989	0.5533	48.4700	100.0000
	Baboon	7.4745	7.9894	0.5149	50.5200	99.9870
Ma [12]	Lena	7.4456	7.4470	0.0014	48.4400	100.0000
	Baboon	7.4745	7.4824	0.0079	50.3300	99.9794
Nguyen [14]	Lena	7.4456	7.4456	0.0000	21.0600	99.3622
	Baboon	7.4745	7.4745	0.0000	20.8800	99.3404
Zheng [31]	Lena	7.4456	7.4476	0.0020	30.9900	99.9999
	Baboon	7.4745	7.4754	0.0009	30.8900	99.8990
Li [9]	Lena	7.4456	7.4456	0.0000	48.4500	99.9999
	Baboon	7.4745	7.4933	0.0188	50.3400	99.9851
Li [11]	Lena	7.4456	7.9979	0.5523	48.4465	99.9997
	Baboon	7.4745	7.9984	0.5239	50.3373	99.9862
Malik [13]	Lena	7.4456	7.9975	0.5519	28.6365	99.6297
	Baboon	7.4745	7.9964	0.5219	28.7273	99.6162
Xu [25]	Lena	7.4456	7.9948	0.5492	47.4345	99.8979
	Baboon	7.4745	7.9143	0.4398	49.7656	99.8821

Table 6 Complexity[♯] performance comparison for different schemes over the large-scale natural image set BOSSbase

Schemes	Zheng [31]	Nguyen [14]	Ma [12]	Li [9]	Wu [22]	Li [11]	Xu [25]	Proposed
Time(s)	0.303	0.242	1.061	1.200	0.815	0.605	1.128	1.280

[♯] Matlab platform over Lenovo machine with 8GB RAM and Intel I5 Eight cores 2.30GHz

where $c_1(i, j)$ and $c_2(i, j)$ are the corresponding pixels (the i^{th} row and j^{th} column) of the original image and the encrypted image, respectively. $M \times N$ are the actual size of the image. For a gray-scale image, the theoretical values of NPCR and UACI are usually 100 % and 33.46 % [19], respectively.

In order to show the superior performance of proposed scheme, five RDH schemes are selected to give the comparison results. In this experiment, we use the Lena and Baboon to test the security performance and the corresponding results are shown in Table 5. For proposed scheme, the entropy and UACI of encrypted image is closer to the standard theoretical value than other schemes, no matter whether the testing image is smooth or texture. The NPCR of encrypted image are significantly higher than that of other schemes, especially higher than the schemes without stream cipher encryption [14, 31]. This result means that the proposed scheme has a superior security performance than other schemes. In addition, we should note that since the bit planes are implemented plane cycling-XOR process and the vacated rooms are transmitted to the LSB plane, it must be ensured that the additional data are accurate during transmission, otherwise, the original image may not be decrypted.

4.5 Complexity analysis

Additionally, we also test the complexity of proposed scheme. In order to make a specific comparison of complexity performance, we test the average running time over BOSSbase v1.01 database. Seven existing RDH schemes, Zheng's scheme [31], Ma's scheme [12], Nguyen's scheme [14], Li's scheme [9], Wu's scheme [22], Li's scheme [11] and Xu's scheme [25], are used to provide the comparison results, which are shown in Table 6. Our scheme may consume slight more time than other methods in practical application. This is because our scheme performs bit plane cycling-XOR procedure, resulting in a high time-consuming, e.g., the average gains are 0.50-0.80 seconds comparing with other schemes. Nevertheless, this time-consuming should not be a concern at all, because this time cost is basically negligible in practical application scenario and in this case high-performance computing equipment may provide a better choice. Moreover, we should note that the plane cycling-XOR procedure essentially plays the role of pixel encryption, and thus improves the scrambling effectiveness of encrypted image. Accordingly, the proposed scheme has a higher security performance comparing with other schemes.

5 Conclusions and future works

In this paper, we studied the reversible data hiding problem in encrypted images. Since existing works is difficult to attain a good balance among good visual quality, large embedding capacity and high security performance for encrypted images, we addressed this problem by combining median prediction model and bit plane cycling XOR. We estimated the MSB

of each pixel by designing a median predictor and generated a prediction error map to mark the pixels whose MSB bits are predicted incorrectly. Then, the plane cyclic exclusive OR from LSB plane to MSB plane is implemented to vacate the LSB plane. Subsequently, data hider embedded the additional data into the vacated LSB plane after the image is encrypted by the stream cipher, and delivered the encrypted image containing the additional data to the recipient, who performed the separable operations of data extraction, image decryption and image recovery according to the obtained keys. Comprehensive experiments demonstrate that compared with existing methods, our scheme can get a better balance among good visual quality, large embedding capacity and high security performance.

Although proposed scheme has been verified to achieve a superior performance than some state-of-the-arts, we should note that our method is actually not sensitive to the level of texture complexity of image. In other words, our method has approximate performance regardless of whether it is a textured image or a non-textured image. This is mainly because our scheme uses the plane cyclic exclusive OR. In essential, this process transfers the embedding room of the MSB plane to the LSB plane, and the remaining thing only needs to predict the MSB plane. Meanwhile, the median predictor can achieve a very high accuracy rate for MSB prediction so that the proposed scheme can vacate approximately the same embedding room regardless of whether it is a texture image or a smooth image, leading to insensitivity to image texture.

In the future, we plan to further improve our scheme in two directions. First, the prediction model can be optimized by involving more adjacent pixels. Second, future works should look for a better reversible strategy by introducing deep adversarial learning. The above two issues are left as our future work.

Acknowledgements The authors would like to thank the editors and anonymous reviewers for their valuable suggestions and comments.

Author Contributions Fengyong Li designed the proposed algorithm and drafted the article. Hengjie Zhu designed and conducted the subjective experiments and tested the proposed algorithm. Chuan Qin offered useful suggestions and modified the article. All authors read and approved the final article.

Funding This work was supported by the National Natural Science Foundation of China (No. U1936213), Natural Science Foundation of Shanghai (No. 20ZR1421600) and Research Fund of Guangxi Key Lab of Multi-source Information Mining & Security (No. MIMS21-M-02).

Declarations

Conflict of Interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. BOSSbase v1.01 (2020). http://dde.binghamton.edu/download/. Accessed March 2020.
2. Cao X, Du L, Wei X et al (2016) High capacity reversible data hiding in encrypted images by patch-level sparse representation. IEEE Trans Cybern 46(5):1132–1143
3. Chen K, Chang C (2019) Error-free separable reversible data hiding in encrypted images using linear regression and prediction error map. Multimed Tools Appl 78(22):31441–31465
4. Golomb S (1966) Run-length encodings. IEEE Trans Inf Theory 12(3):399–401
5. Huang D, Wang J (2020) High-capacity reversible data hiding in encrypted image based on specific encryption process. Signal Process Image Commun 80:115632

6. Khosravi MR, Yazdi M (2018) A lossless data hiding scheme for medical images using a hybrid solution based on IBRW error histogram computation and quartered interpolation with greedy weights. Neural Comput Applic 30(7):2017–2028
7. Kim C, Yang CN, Leng L (2020) High-capacity data hiding for ABTC-EQ based compressed image. Electronics 9(4):644
8. Li F, Wu K, Qin C et al (2020) Anti-compression JPEG steganography over repetitive compression networks. Signal Process 107454:170
9. Li Q, Yan B, Li H et al (2018) Separable reversible data hiding in encrypted images with improved security and capacity. Multimed Tools Appl 77:30749–30768
10. Li F, Zhang L, Wei W (2020) Reversible data hiding in encrypted binary image with shared pixel prediction and halving compression. EURASIP J Image Video Process 2020(33):1–21
11. Li F, Zhu H, Yu J, Qin C (2021) Double linear regression prediction based reversible data hiding in encrypted images. Multimed Tools Appl 80:2141–2159
12. Ma K, Zhang W, Zhao X et al (2013) Reversible data hiding in encrypted images by reserving room before encryption. IEEE Trans Inf Forensics Secur 8(3):553–562
13. Malik A, Wang H, Chen Y, Khan A (2020) A reversible data hiding in encrypted image based on prediction-error estimation and location map. Multimed Tools Appl 79:11591–11614
14. Nguyen T, Chang C, Chang W (2016) High capacity reversible data hiding scheme for encrypted images. Signal Process Image Commun 44:84–91
15. Puteaux P, Puech W (2018) An efficient MSB prediction-based method for high-capacity reversible data hiding in encrypted images. IEEE Trans Inf Forensics Secur 13(7):1670–1681
16. Qi W, Li X, Zhang T, Guo Z (2019) Optimal reversible data hiding scheme based on multiple histograms modification. IEEE Trans Circuits Syst Video Technol 30(8):2300–2312
17. Qin C, Ji P, Chang C et al (2018) Non-uniform watermark sharing based on optimal iterative BTC for image tampering recovery. IEEE Multimedia 25(3):36–48
18. Shi Y, Li X, Zhang X, Wu H, Ma B (2016) Reversible data hiding:advances in the past two decades. IEEE Access 4:3210–3237
19. Tang Z, Xu S, Yao H et al (2019) Reversible data hiding with differential compression in encrypted image. Multimed Tools Appl 78:9691–9715
20. Tian J (2003) Reversible data embedding using a difference expansion. IEEE Trans Circuits Syst Video Technol 13(8):890–896
21. Wang Y, Cai Z, He W (2020) High capacity reversible data hiding in encrypted image based on intra-block lossless compression. IEEE Trans Multimedia 23:1466–1473
22. Wu H, Li F, Qin C, Wei W (2019) Separable reversible data hiding in encrypted images based on scalable blocks. Multimed Tools Appl 78(18):25349–25372
23. Wu X, Sun W (2014) High-capacity reversible data hiding in encrypted images by prediction error. Signal Process 104(6):387–400
24. Xiao M, Li X, Ma B, Zhang X, Zhao Y (2020) Efficient reversible data hiding for JPEG images with multiple histograms modification. IEEE Trans Circuits Syst Video Technol 31(7):2535–2546
25. Xu S, Chang CC, Liu Y (2021) A high-capacity reversible data hiding scheme for encrypted images employing vector quantization prediction. Multimed Tools Appl 80:20307–20325
26. Yi S, Zhou Y (2018) Separable and reversible data hiding in encrypted images using parametric binary tree labeling. IEEE Trans Multimedia 21(1):51–64
27. Yin Z, Xiang Y, Zhang X (2019) Reversible data hiding in encrypted images based on multi-MSB prediction and huffman coding. IEEE Trans Multimedia 22(4):874–884
28. Zeng K, Chen K, Zhang W, Wang Y, Yu N (2022) Improving robust adaptive steganography via minimizing channel errors. Signal Process 108498:195
29. Zhang X, Qin C, Sun G (2012) Reversible data hiding in encrypted images using pseudorandom sequence modulation, digital forensics and watermarking, Springer Berlin Heidelberg, 2012
30. Zhang X et al (2012) Separable reversible data hiding in encrypted image. IEEE Trans Inf Forensics Secur 7(2):826–832
31. Zheng S, Li D, Hu D et al (2016) Lossless data hiding algorithm for encrypted images with high capacity. Multimed Tools Appl 75(21):13765–13778

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.