Copyright © 2018 Tech Science Press CMC, vol.56, no.2, pp.299-312, 2018

Reversible Data Hiding in Classification-Scrambling Encrypted-Image Based on Iterative Recovery

Yuyu Chen[1], Bangxu Yin[2], Hongjie He[2], Shu Yan[2], Fan Chen[2, *] and Hengming Tai[3]

Abstract: To improve the security and quality of decrypted images, this work proposes a reversible data hiding in encrypted image based on iterative recovery. The encrypted image is firstly generated by the pixel classification scrambling and bit-wise exclusive-OR (XOR), which improves the security of encrypted images. And then, a pixel-type-mark generation method based on block-compression is designed to reduce the extra burden of key management and transfer. At last, an iterative recovery strategy is proposed to optimize the marked decrypted image, which allows the original image to be obtained only using the encryption key. The proposed reversible data hiding scheme in encrypted image is not vulnerable to the ciphertext-only attack due to the fact that the XOR-encrypted pixels are scrambled in the corresponding encrypted image. Experimental results demonstrate that the decrypted images obtained by the proposed method are the same as the original ones, and the maximum embedding rate of proposed method is higher than the previously reported reversible data hiding methods in encrypted image.

Keywords: Reversible data hiding, image encryption, scrambling encryption, iterative recovery.

1 Introduction

Information security includes many research branches, such as Steganography [Zhang, Qin, Zhang et al. (2018)], information hiding, image Forensics [Wang, Li, Shi et al. (2018)] and Steganalysis [Ma, Luo, Li et al. (2018)]. Reversible image data hiding (RDH), as an important sub-branch of information hiding, is a technique which utilizes the redundant information of digital image to reversibly embed the secret data into it, so that at the receiver side, the embedded data can be correctly extracted and original image can be perfectly recovered [Ni, Shi, Ansari et al. (2006)]. The idea of reversible data hiding in encrypted images (RDH-EI) is inspired by the existing works on image processing in encrypted domain for cloud computing and privacy-preserving applications [Zhang (2011)]. As the name suggests, RDH-EI has all the advantages of both image encryption and RDH. On the one hand, image encryption is a privacy protection technique by transforming the original image into a noise-like encrypted vision. On the other hand, RDH achieves the copyright protection, content integrity authentication and

[1] Mao Yisheng Honors College, Southwest Jiaotong University, Chengdu, 611756, China.

[2] School of Information Science and Technology, Southwest Jiaotong University, Chengdu, 611756, China.

[3] Department of Electrical Engineering, University of Tulsa, Tulsa, OK 74104, USA.

[*] Corresponding Author: Fan Chen. Email: fchen@swjtu.cn.

CMC. doi:10.3970/cmc.2018. 03179

management by reversibly embedding some additional data in the encrypted images. Of course, the data hider can only embed and extract additional data without access to the image content since it is not completely trusted. With the rapid development of cloud computing technology, RDH-EI technique [Shi, Li, Zhang et al. (2016)] has become one of the research hot spots in recent years.

Zhang [Zhang (2011)] proposed a joint RDH-EI method. According to the encryption key, an original image was encrypted by the bitwise exclusive-or (XOR) operation. According to the hiding key, the data-hider flips half of the 3 LSB (least significant bit) in an image block to embed 1 bit data. To further improve the embedding rate (ER), researchers proposed a variety of improved joint RDH-EI methods [Liao and Shu (2015); Qian, Dai, Jiang et al. (2016); Qian and Zhang (2016)]. From the viewpoint of reducing the difficulty of key management, Zhou et al. [Zhou, Sun, Dong et al. (2016)] proposed a joint RDH-EI via public key modulation instead of hiding key, and a two-class SMV classifier was designed for data extraction and image recovery. Compared with the joint RDH-EI methods, the separable RDH-EI methods can perform data extraction and image decryption, respectively. Specifically, according to the marked-encrypted image, the embedded data can be extracted only with hiding key, while the decrypted image similar to the original one can be obtained only with encryption key [Shi, Li, Zhang et al. (2016)]. The existing separable RDH-EI methods can be divided into categories: methods based on vacating room after encryption (VRAE) [Qian, Dai, Jiang et al. (2016); Qian and Zhang (2016); Huang (2016); Yin, Chen, He et al. (2017)] and methods based on reversing room before encryption (RRBE) [Ma, Zhang, Zhao et al. (2013); Cao, Du, Wei et al. (2016); Xu and Wang (2016); Yin, Chen, He et al. (2017)]. The VRAE methods have low embedding rate (ER) and may lead to errors in data extraction and image recovery. This is due to the fact that it is difficult to losslessly create room for data hiding since the entropy of an encrypted image is maximized. On the contrary, the RRBE methods have high ER and maintains that both data extract and image recovery are lossless.

Most existing RDH-EI techniques have focused on the basis of two conflicting measures, i.e. the ER and the quality of decrypted images. The research on the security aspect of RDH-EI is relatively weak. For example, in Xu’s method [Xu and Wang (2016)], a special XOR encryption method was designed to protect interpolation-error of non-sample pixels, which made it possible to embed more additional data into the interpolation-error by the traditional RDH methods. However, there is obvious information leakage in encrypted images generated by Xu’s method since the pixels with interpolation-errors near to 0 in the encrypted image were not encrypted [Xu and Wang (2016)]. Furthermore, most existing RDH-EI techniques [Qian, Dai, Jiang et al. (2016); Qian and Zhang (2016); Ma, Zhang, Zhao et al. (2013); Cao (2016); Xu and Wang (2016)] adopted a bit-wise XOR to generate encrypted image. Though values of encrypted pixels randomly distributed in the range of (0, 255), the position of original pixels had not changed, which leads to security risk in encrypted image [Yin, Chen, He et al. (2017)]. Recently, based on the spatial redundancy that characterizes natural images, Khelifi [Khelifi (2018)] proposed a cipher-text only attack (COA) to further highlight the weakness of a bit-wise XOR encryption. Using COA attack, 480 encrypted images generated by the same encryption key are sufficient to accurately estimate the keystream used to encrypt 5 most significant bit (MSB) planes [Khelifi (2018)]. This enables the attacker to reconstruct images with high visual quality in terms of above 42 dB PSNR (Peak Signal to Noise Ratio). As a result, there are serious security risks with this type of RDH-EI methods since the primary aim of encryption is to protect the image content. To improve the security, Yin et al. [Yin, Chen, He et al. (2017)] proposed a separable RDH-EI method with classification permutation was designed to protect both pixel-value and pixel-location, in which a classification permutation encryption combining with the bit-wise XOR encryption. Yin's method [Yin, Chen, He et al. (2017)] is not vulnerable to the COA due to the pseudo-randomly permuting pixels. However, it has an extra burden of storage and transmission since the pixel-type-mark (PTM) was also shared between the content-owner and receiver. Furthermore, the quality of decrypted image decreases with increasing ER.

To address the above problems, this work analyzes the reason why Yin's scheme has the above problems. On the basis of ensuring the security of the RDH-EI scheme, the main innovation includes two aspects: (1) a PTM generation method based on block compression is designed to reduce the extra burden of key management and transfer; (2) an iterative recovery strategy is proposed to reconstruct the original image without lossless according to the marked decrypted images, even if the ER reaches a maximum value. The proposed RDH-EI scheme has the same high security as the Yin's method [Yin, Chen, He et al. (2017)] because the XOR-encrypted pixels are scrambled again. Experimental results demonstrate that the ER and quality of decrypted images of our proposed scheme outperforms the previously reported RDH-EI methods.

2 Analysis of Yin's scheme

Same to existing separable RDH-EI methods, an original image X with a size of m×n pixels is assumed to be uncompressed format and each pixel $x_{i,j}$ with gray value falling into (0, 255), where $1 \leq i \leq m$ and $1 \leq j \leq n$. And Yin's scheme [Yin, Chen, He et al. (2017)] contains five phases: Image encryption, data hiding, data extraction, image decryption and image recovery. For data hiding and data extraction of Yin's scheme, the additional data is embedded into the MSB of pixels in the specified area of encrypted images, which is similar to the RDH-EI based on RRBE. The hiding key is used to encrypt the addition data and select the carrying pixels in the specified encrypted image area. As a result, Yin's scheme enables fast embedding and lossless extraction of additional data in encrypted image. The innovations of Yin's method lie in three aspects as following: Image encryption, Image decryption and Image recovery. The design objective of the proposed method is: Only by the encryption key, the decrypted image with the same as the original one can be obtained, so the image recovery phase is omitted. In the following, the image encryption and decryption parts of Yin's RDH-EI method are introduced and analyzed separately.

2.1 Image encryption of Yin's scheme

In Yin's image encryption, all pixels in the original image are firstly classified into smooth pixels and non-smooth ones before image encryption. The binary matrix T with the same size of original image, which is called as the pixel-type-mark (PTM), is used to record pixel type and obtained according to the 3×3 neighborhood of it,

$$\mathrm{t}_{i,j}=\begin{cases}1 \quad , \quad i=1,m \ or \ j=1,n \\ 1 \quad , \quad MSB \ of \ all \ pixels \ in \ \Delta_{i,j}^{X} \ are \ not \ same \\ 0 \quad , \quad MSB \ of \ all \ pixels \ in \ \Delta_{i,j}^{X} \ are \ same\end{cases} \tag{1}$$

where $\Delta_{i,j}^{X}$ is a 3×3 neighborhood centered on the pixel $x_{i,j}$ in the original image. From other viewpoint, if $t_{i,j}=0$, the corresponding pixel $x_{i,j}$ in the original image is smooth pixel; otherwise, the pixel $x_{i,j}$ is non-smooth one. After pixel classification, according to the encryption key, the encrypted image is produced by scrambling the smooth pixels and the non-smooth ones in the XOR-encrypted image obtained by the bitwise XOR, respectively. The detailed image encryption method refers to Yin's scheme [Yin, Chen, He et al. (2017)].

In encrypted image generated by Yin's RDH-EI scheme, both pixel-value and pixel-position are protected, which makes it less vulnerable to COA attack. Furthermore, the classification permutation makes it possible to discriminate the smooth pixels in the encrypted image without knowing the original image content. It should be noted that the MSB of smooth pixels, which are arranged in the front of the encrypted image, is used to embed the additional data. According to formula (1), the MSB of a smooth pixel is the same as that of its 8-neighborhood pixel. Therefore, the MSB of the smooth pixel can be corrected based on its surrounding pixels' MSB to obtain a high-quality decrypted image. This is also the theoretical basis for improving decryption quality in Yin's scheme.

2.2 Image decryption of Yin's scheme

According to the marked-encrypted image E and the encryption key including the corresponding PTM T, the marked-decrypted image $D^0=\{\mathrm{d}_{i,j}^{0}|1\leq i\leq m,\ 1\leq i\leq n\}$ can be obtained by performing the inverse process of the encryption process of Yin et al. [Yin, Chen, He et al. (2017)]. Note that there are some pixels whose MSB may be error in the marked-decrypted image D^0 due to the additional data embedding in encrypted image. To obtain the decrypted image with good quality, the decrypted image $D=\{d_{i,j}|1\leq i\leq m,\ 1\leq i\leq n\}$ is firstly initialized to the marked-decrypted image D^0, and then the value of the pixel $d_{i,j}$ such that $t_{i,j}=0$ is adjusted by,

$$\mathrm{d}_{i,j}=\begin{cases}128+mod\left(d_{i,j},128\right), \ if \ \sum_{\delta\in\Delta_{i,j}^{D^0}}\left\lfloor\frac{\delta}{128}\right\rfloor\geq 4 \\ mod\left(d_{i,j},128\right), \qquad\qquad otherwise\end{cases} \tag{2}$$

where $\Delta_{i,j}^{D^0}$ is a 3×3 neighborhood centered on the pixel $\mathrm{d}_{i,j}^{0}$ in the marked-decrypted image D^0.

2.3 Problem analysis of Yin's scheme

Yin's scheme improved the security by scrambling the XOR-pixels and enlarged the EC by the neighborhood prediction. When the amount of embedded data is not large, the decrypted image with high quality can be obtained by the neighborhood statistics. However, the following problems exist in Yin's method.

(1) The size of PTM T is the same as that of the original image. The PTM can improve

the security of the RDH-EI method as it is part of the encryption key and is different for different images. The PTM must be saved and translated through a secure channel, which will increase both the bandwidth burden of the secret channel and the difficulty of key management. So, how to effectively reduce the amount of PTM data is a key problem.

(2) As we can know from Yin's image decryption, the MSB of the smooth pixels is adjusted according to the statistical features of its 3×3 neighborhood. The pixel can restore the original value correctly according to the formula (2) only if there are few changed pixels in the 3×3 neighborhood of it during data hiding phase. On the contrary, it is difficult to restore the MSB of the pixel if most of 8 neighbor pixels adjacent to it have changed, which it is easy to know from binomial distribution knowledge. Once there is one error recovery of smooth pixels, the quality of the decrypted image will drop rapidly.

To address the problems mentioned above, this work proposes a separable RDH-EI method based on iterative recovery. In the proposed scheme, the PTM does not need to be transmitted over the secret channel, and the decrypted image is the same as the original one even if the maximum ER is reached.

3 Proposed scheme

The main contributions of this work include two aspects. The block-compression based PTM generation method is designed for easy key management, and the iterative recovery method is proposed to improve the quality of the decrypted image.

3.1 Generating the block-compression based PTM

It can be known from formula (1), the size of PTM T is the same as that of the original image. If the PTM cannot be transferred as part of the encryption key, it must be hidden in the encrypted image. Thus, the amount of PTM must be decreased. Thanks to the local correlation of natural images, the block-compression based PTM (BC-PTM) is generated, which can effectively reduce the amount of PTM data without significantly increasing the proportion of smooth pixels.

All border pixels of PTM T generated by (1) are set to 0, and the preprocessed PTM T is divided into non-overlapping blocks of m_b×n_b pixels, expressed as,

$$T = \{T_i | i = 1,2, \dots , N\} \tag{3}$$

Where N be the number of blocks in the PTM T and can be expressed as,

$$N = \lceil m/m_b \rceil \times \lceil n/n_b \rceil \tag{4}$$

From (4), it is easy to know that the size of the original image may not be divisible by the block size. Let N_b be the number of pixels in each block, i.e. $N_b = m_b \times n_b$, according to the raster scanning order, each m_b×n_b block T_i can be expressed as,

$$T_i = \begin{bmatrix} t_{i,1} & t_{i,2} & \dots & t_{i,n_b} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & t_{i,N_b} \end{bmatrix} \tag{5}$$

In this work, we use a binary matrix $B=\{b_i|1\leq i\leq N\}$ to represent the block-type-mark (BTM) of an original image, and it is generated by block-compression. To obtain the recovered image with high quality, the corresponding pixel b_i can only be 0 if all pixels in block T_i are all 0 s. That is,

$$b_i = \begin{cases} 0 & , \quad t_{i,j} = 0, \forall j \in [1, N_b] \\ 1 & , \quad otherwise \end{cases} \tag{6}$$

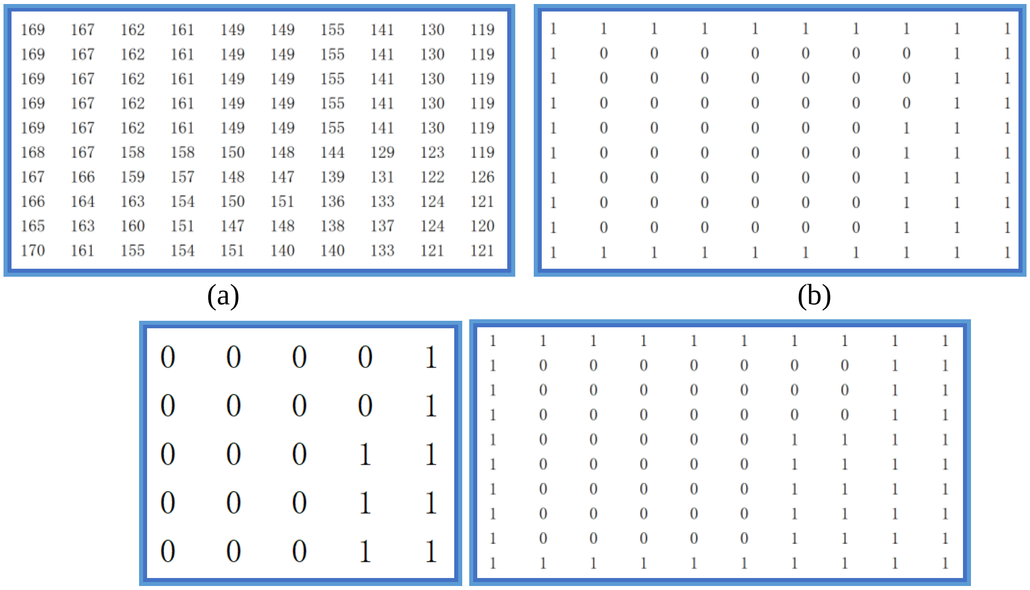
According to the BTM B, the BC-PTM with $m\times n$ pixels, denoted as P, can be reconstructed by performing the inverse of block-compression coding. Specifically, all pixels in block P_i with size of $m_b\times n_b$ are set to 0 if the corresponding b_i is 0, and all pixels in block P_i are set to 1 if the corresponding b_i is 1. Note that, it is possible that the size of BC-PTM P is not smaller than that of original image. The BC-PTM P with size $m\times n$ pixels can be obtained by the bottom extra lines and the right extra column. At last, all boundary pixels of BC-PTM P with size of $m\times n$ pixels are set to 1 since the 3×3 neighborhood centered on boundary pixel are incomplete.

In summary, the BTM B, which is generated by block-compression coding, can be used to reconstruct the BC-PTM P. The amount of BTM is reduced to 1/(mb×nb) of the PTM amount, which makes it possible to hide it in encrypted image. Of course, there are no more smooth pixels in BC-PTM P than in PTM T. Let α_T and α_P be the proportion of smooth pixels in the PTM T and BC-PTM P, respectively. That is,

$$\begin{cases} \alpha_T = 1 - \left(\sum_i \sum_j t_{i,j} / (m \times n)\right) \\ \alpha_P = 1 - \left(\sum_i \sum_j p_{i,j} / (m \times n)\right) \end{cases} \tag{7}$$

Obviously, $\alpha_P \leq \alpha_T$. Fortunately, the value of α_P is not too much smaller than that of α_T due to the local correlation of natural images.

To intuitively understand the relationship between PTM and BC-PTM, Fig. 1 shows an example of generating BC-PTM process, where Fig. 1(a) is an original image with size of 10×10 pixels. According to formula (1), we can obtain the PTM T of Fig. 1(a), as shown in Fig. 1(b), where the number of smooth pixels and non-smooth ones are 51 and 49, respectively. The proportion of smooth pixels in the PTM T is 51/100, i.e. $\alpha_T = 0.51$. Let the block size be 2×2 pixels, the BTM B can be achieved according to formula (6), as shown in Fig. 1(c). Note that, all border pixels of PTM T are set to 0 before the BTM is generated. As can be seen from Fig. 1(c), the number of pixels in the BTM is 25. Fig. 1(d) shows the BC-PTM P, in which the proportion of smooth pixels is 46/100, i.e. $\alpha_P = 0.46$. If the BTM is hidden in smooth pixels of encrypted image, the number of smooth pixels that can be used to embed additional data is 46-25=21.



169	167	162	161	149	149	155	141	130	119
169	167	162	161	149	149	155	141	130	119
169	167	162	161	149	149	155	141	130	119
169	167	162	161	149	149	155	141	130	119
169	167	162	161	149	149	155	141	130	119
168	167	158	158	150	148	144	129	123	119
167	166	159	157	148	147	139	131	122	126
166	164	163	154	150	151	136	133	124	121
165	163	160	151	147	148	138	137	124	120
170	161	155	154	151	140	140	133	121	121

(a)

1	1	1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	0	1	1
1	0	0	0	0	0	0	0	1	1
1	0	0	0	0	0	0	0	1	1
1	0	0	0	0	0	0	1	1	1
1	0	0	0	0	0	0	1	1	1
1	0	0	0	0	0	0	1	1	1
1	0	0	0	0	0	0	1	1	1
1	0	0	0	0	0	0	1	1	1
1	1	1	1	1	1	1	1	1	1

(b)

0	0	0	0	1
0	0	0	0	1
0	0	0	1	1
0	0	0	1	1
0	0	0	1	1

(c)

1	1	1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	0	1	1
1	0	0	0	0	0	0	0	1	1
1	0	0	0	0	0	0	0	1	1
1	0	0	0	0	0	1	1	1	1
1	0	0	0	0	0	1	1	1	1
1	0	0	0	0	0	1	1	1	1
1	0	0	0	0	0	1	1	1	1
1	0	0	0	0	0	1	1	1	1
1	1	1	1	1	1	1	1	1	1

(d)

Figure 1: An example of generating BC-PTM process (a) Original image with 10×10 pixels (b) PTM T with 0.51 proportion of smooth pixels (c) BTM B with 5×5 pixels, and (d) BC-PTM P with 0.46 proportion of smooth pixels

In the proposed scheme, the BTM is hidden in the smooth pixels of the encrypted image based on the encryption key, which makes the BTM not be transmitted through the secret channel. The receiver can extract the BTM from the marked-encrypted image according to the encryption key and reconstruct the BC-PTM. On the other side, to let data-hider know how many pixels in encrypted image can be used to embed the additional data, we embed the binary-code of the number of smooth-pixels into the MSB of the first $\lceil log_2(m \times n) \rceil$ pixels in the encryption image. As a result, the maximum ER of the proposed RDH-EI scheme is,

$$E_{max} = \alpha_P - \left(\frac{1}{(m_b \times m_b)}\right) - \left(\frac{\lceil log_2(m \times n) \rceil}{(m \times n)}\right) \tag{8}$$

3.2 Iterative recovery

In the proposed scheme, only by the encryption key, the EC-PTM *P'* is first reconstructed according to the BTM extracted from the marked-encrypted image. And then the marked-decrypted image D^0 is obtained. According to the marked-decrypted image D^0 and EC-PTM *P'*, the goal of the proposed iterative recovery is to obtain the decrypted image which is the same as the original one.

Because the hiding key is unknown, there is no way to know whether the smooth pixels were changed in the data hiding or image encryption phases. To get the same decrypted image as the original image, the MSB of all the smooth pixels must be estimated correctly. Specifically, for $\forall p'_{i,j} = 0$, the MSB of corresponding pixel $d^0_{i,j}$ in D^0 must be recovered. According to the formula (2), the MSB of smooth pixel is the same as that of 8-neighborhood pixels of it. For ease of description, 8 pixels adjacent to pixel $\mathrm{p}'_{i,j}$ in *P'* are represented as $\{\delta^k_{p'_{i,j}} | k = 1,2,\ldots,8\}$, and 8 pixels adjacent to pixel $d^0_{i,j}$ in the D^0 are represented as $\{\delta^k_{d^0_{i,j}} | k = 1,2,\ldots,8\}$. For each smooth pixel, if there are non-smooth pixels in the 8 pixels that are adjacent to it, the MSB of it must be correctly obtained by the MSB of the non-smooth. Let *D* represent the decryption image, and the binary matrix *F* indicates whether the MSB of the corresponding pixel in *D* is correct. The above process is described by the formula as,

$$f_{i,j} = 1,\ if\ \left(p'_{i,j} = 0\right)\ \&\ \left(\sum_{k=1}^{8} \delta^k_{p'_{i,j}} > 0\ \right) \tag{9}$$

$$d_{i,j} = 128 \times \left\lfloor \delta^k_{d^0_{i,j}}/128 \right\rfloor + mod\left(d^0_{i,j}, 128\right), if\ \ \delta^k_{p'_{i,j}} = 1 \tag{10}$$

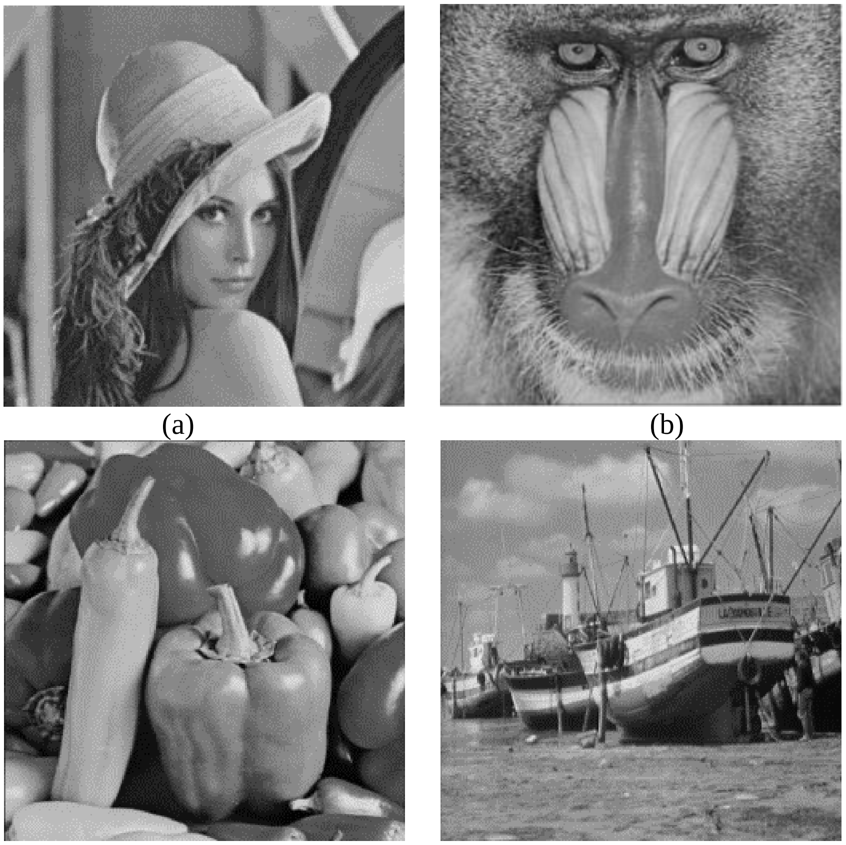
The optimization process based on iterative recovery takes a certain number of iterations to obtain the decrypted image *D*. Denote N_{max} as the maximum number of iterations. The proposed image decryption based on iterative recovery is summarized in Algorithm 1.

```
Algorithm 1: Proposed iterative recovery for image decryption
Input: The marked-decrypted image $D^0$, the extracted EC-PTM P'
Output: Final decrypted image $D$,  Number of iterative $N_{max}$
Initialization: $D$=$D^0$; Mmax=0; $F$=P'
Iterative optimization:
  1)  for i=1,2…,m , for j=1,2…,n do
          If each smooth pixel and 8 pixels adjacent to it exist non-smooth
              pixel,    update $D$&$F$ according to formula (9) and (10).
          else
             not update
       end
      Updata $D^0$=D;  P'=F;
      $N_{max}$++
  2)  repeat 1) until all of the elements in F become 1
Result: Obtain the decrypted image $D$, Number of iterative $N_{max}$
```

4 Experiment results

In all of our experiments, the additional data are randomly generated. Four images with the size of 512×512 pixels are utilized, which are popularly used in the testing the efficiency of RDH schemes by other researchers. The tested images including Lena, Baboon, Peppers and Boat are shown in Fig. 2. The encrypted image obtained by proposed scheme looks like random noise image, since pixel values in encrypted image are randomly distribute in the range [0, 255], and the positions had also been changed.

The proposed method is completely reversible in terms of both data extraction and image recovery. Thus, only the embedding rate and visual quality of the decrypted image are concerned for the following experiments.



(a) (b)

(c) (d)

Figure 2: Tested images (a) Lena, (b) Baboon, (c) Peppers, (d) Boat

4.1 Performance analysis

The ER of proposed scheme related to the number of smooth pixels and the length of BTM. With the increasing of block size, the length of BTM gets smaller while the smooth pixels decrease. To illustrate the relationship among the number of smooth pixels, the block size and the ER, the test Lena image, shown as in Fig 2(a), was used as the original image in this experiment. Fig. 3(a) is the PTM of Lena image, where the number of the smooth pixels (the black ones) in them are 219811. Fig. 3(b) is the BC-PTM when the block size is 2×2 pixels, where the number of the smooth pixels is 206612. According to the formula (8), the maximum ER of Lena is about 0.54 bpp. As the block size increases to 4×4 pixels, the BC-PTM is shown in Fig. 3(c), in which the number of the smooth pixels reduced to 185776. In this case, the maximum ER of Lena is about 0.64 bpp.

In the proposed image decryption based on iteration recovery, the number of iterations depends on the image content and the block size, regardless of the ER. Taking the block of size 4×4 as an example, the number of iterations of Lena is 46. Figs. 3(d-f) show the decrypted images after the 1st, 20th and 30th iteration recovery. Among them, there are different degrees of noise in the decrypted image with different iterations. This is because the corresponding pixels are smooth pixels and the MSBs of them are changed during the image encryption or data hiding. With the increase of iteration number, more and more changed pixels are recovered, reducing noise in the decrypted image. When the iteration recovery is completed, the decrypted image, which is the same as the original one, is obtained.

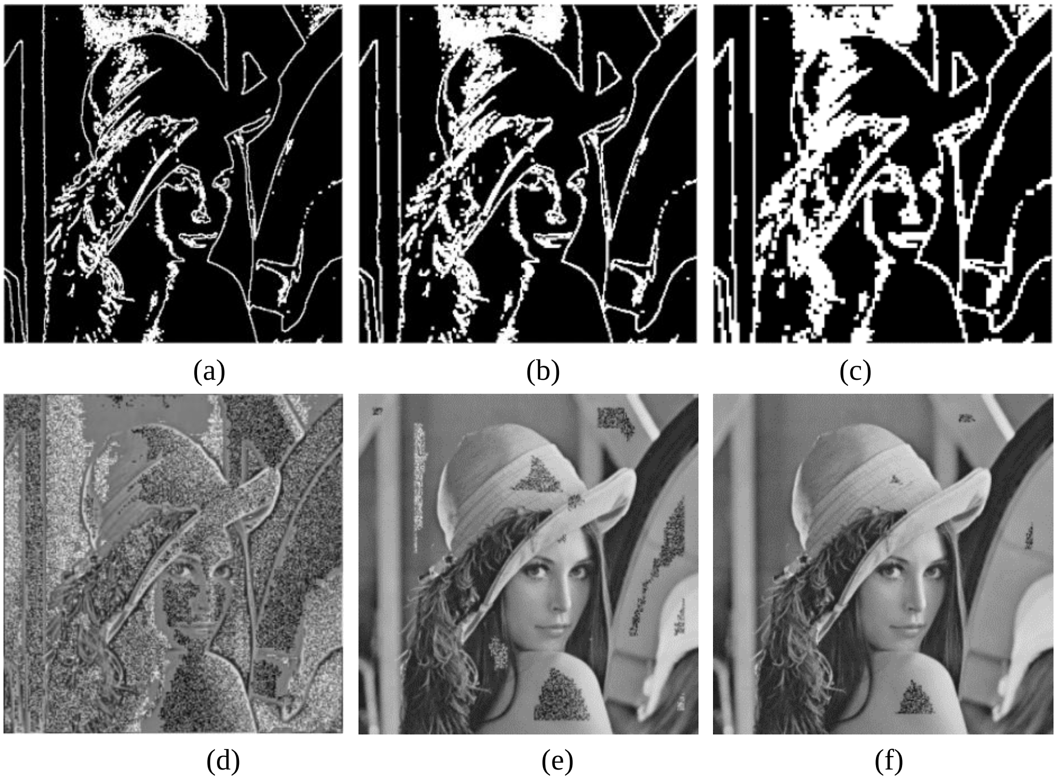


Figure 3: Proposed BC-PMT and decryption process (a) PTM, (b) BC-PTM of size 2×2 pixels, (c) BC-PTM of size 4×4 pixels, (d) the 1st, (e) the 20th, (f) the 30th

4.2 Performance comparison

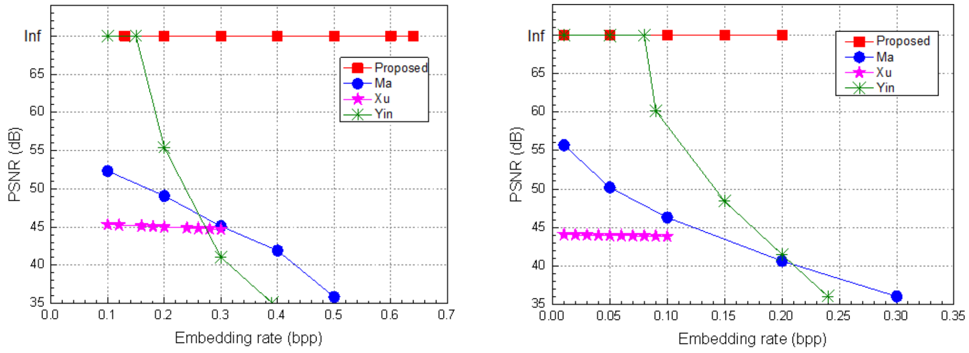
In this subsection, proposed method and three state-of-the-art RRBE methods, i.e. Ma's method [Ma, Zhang, Zhao et al. (2013)], Xu's method [Xu and Wang (2016)] and Yin's method [Yin, Chen, He et al. (2017)] are used in comparisons. The experiments are about the maximum ER and the visual quality of decrypted image. The experiment about the maximum ER is first conducted, and the comparison results are given in Tab. 1. Note that the maximum ER is under the condition that the PSNR of decrypted image is larger than 35 dB. As can be seen from Tab. 1, for the proposed scheme, the maximum ER of Lena, Baboon, Peppers and Boat are 0.64 bpp, 0.20 bpp, 0.69 bpp and 0.52 bpp, respectively. We can see that the smoother the original image is, the higher ER of the proposed method is. Compared with Ma et al. [Ma, Zhang, Zhao et al. (2013)], Xu et al. [Xu and Wang (2016)] and Yin et al. [Yin, Chen, He et al. (2017)] methods, proposed method has the highest embedding rate on image Lena, Peppers and Boat. For Baboon image, the maximum ER of Yin's method is higher than that of proposed method. However, PSNR of the decrypted image obtained by Yin's method is about 35 dB in this case.

Table 1: Maximum EC comparison between proposed, Ma, Xu and Yin methods (bpp)

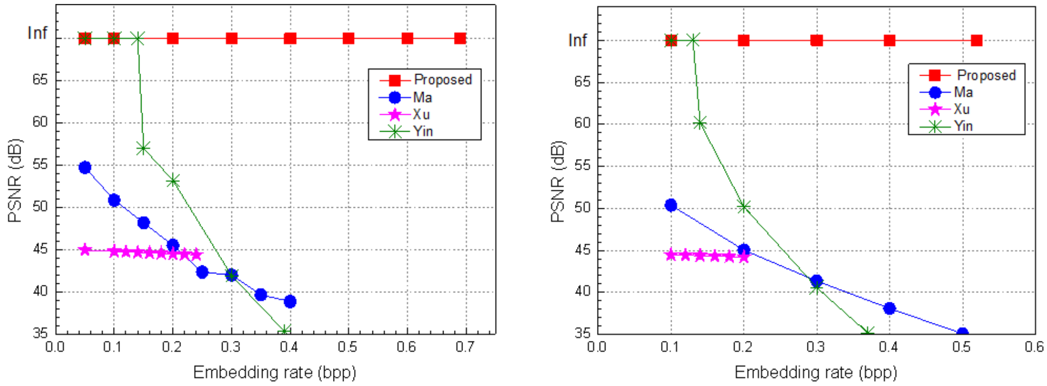
Images	Ma	Xu	Yin	Proposed
Lena	0.50	0.30	0.42	0.64
Baboon	0.30	0.11	0.24	0.20
Peppers	0.40	0.24	0.39	0.69
Boat	0.50	0.20	0.37	0.52

To further verify the performance of proposed method, the experiment about the quality of decrypted image is conducted, and the comparison results are shown in Fig. 4. For the four test images, the proposed method achieved lossy decryption since the PSNR of the decrypted image with different ER is infinite. Yin's method can realize lossless decryption at low ER, while both Ma's method and Xu's method can only realize lossy decryption. As can be seen from Fig. 4, the proposed scheme has the largest ER and the best decryption image quality for Lena, Peppers and Boat images. For texture Baboon image, the maximum ER of the proposed scheme is slightly less than that of Ma and Yin's methods.

We also test the maximum ER of 100 test images. Fig. 5(a) shows the distribution of the maximum ER of the four RDH-EI methods. For each method, we calculate the average ER of 100 images and the average PSNR of decrypted images, as shown in Fig. 5(b). Here the PSNR of each single decrypted image is calculated under its maximum embedding rate. As can be seen from Fig. 5(b) that the average ER of proposed method is higher than 0.5 bpp, while those of the other three comparison methods are less than 0.4 bpp. Moreover, the average PSNR of decrypted images generated by proposed method is inf dB, which is the highest.



(a) (b)

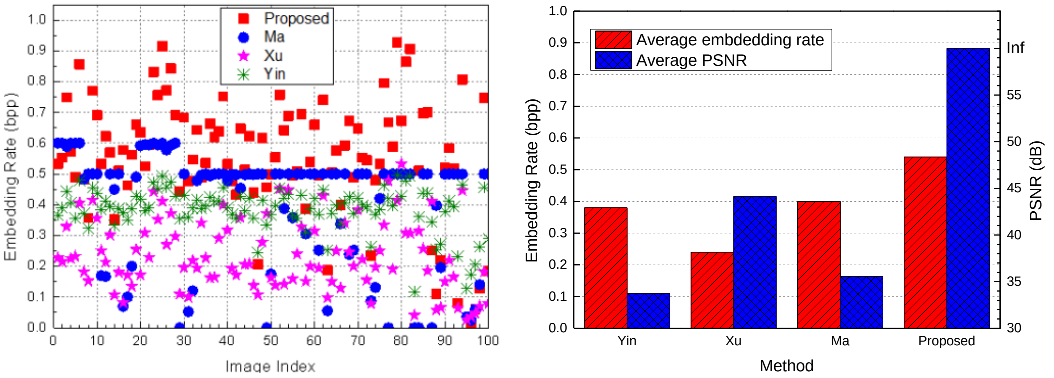


(c) (d)

Figure 4: Decrypted image quality comparison: (a) Lena, (b) Baboon, (c) Peppers, (d) Boat

5 Conclusion and future work

This work proposed a separable reversible data hiding scheme in encrypted image based on iterative recovery. Under the condition that ensure the security of encrypted images, the proposed method allows the original image to be obtained only with the encryption key. The extra burden of key management and transfer are effectively reduced by embedding the block-type-mark, which is used to reconstruct the block-compression based pixel-type-mark, into the encrypted images. Experimental results demonstrate the effectiveness and feasibility of the proposed scheme. In the future, how to increase the embedding rate of texture images should be studied.



(a) (b)

Figure 5: Maximum ER and PSNR of decrypted image for 100 images (a) Maximum ER distribution, (b) Average ER and average PSNR of decrypted images

Acknowledgement: The research is supported by the National Natural Science Foundation of China (61461047, U1536110).

References

Cao, X.; Du, L.; Wei, X.; Dan, M.; Guo, X. (2016): High capacity reversible data hiding in encrypted images by patch-level sparse representation. *IEEE Transactions on Cybernetics*, vol. 46, no. 5, pp. 1132-1143.

Huang, F.; Huang, J.; Shi, Y. (2016): New framework for reversible data hiding in encrypted domain. *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 12, pp. 2777-2789.

Khelifi, F. (2018): On the security of a stream cipher in reversible data hiding schemes operating in the encrypted domain. *Signal Processing*, vol. 143, pp. 336-345.

Liao, X.; Shu, C. (2015): Reversible data hiding in encrypted images based on absolute mean difference of multiple neighboring pixels. *Journal of Visual Communication and Image Representation*, vol. 28, pp. 21-27.

Ma, K.; Zhang, W.; Zhao, X.; Yu, N.; Li, F. (2013): Reversible data hiding in encrypted images by reserving room before encryption. *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 3, pp. 553-562

Ma, Y.; Luo, X.; Li X.; Bao, Z.; Zhang, Y. (2018): Selection of rich model steganalysis features based on decision rough set α-positive region reduction. *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1.

Ni, Z.; Shi, Y. Q.; Ansari, N.; Su, W. (2006): Reversible data hiding. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 3, pp. 354-362.

Qian, Z.; Dai, S.; Jiang, F.; Zhang, X. (2016): Improved joint reversible data hiding in encrypted images. *Journal of Visual Communication and Image Representation*, vol. 40, pp. 732-738.

Qian, Z.; Zhang, X. (2016): Reversible data hiding in encrypted images with distributed source encoding. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 4, pp. 636-646.

Shi, Y. Q.; Li, X.; Zhang, X.; Wu, H.; Ma, B. (2016): Reversible data hiding: Advances in the past two decades. *IEEE Access*, vol. 4, pp. 3210-3237.

Wang, J.; Li, T.; Shi, Y. Q.; Lian, S.; Ye, J. (2018): Forensics feature analysis in quaternion wavelet domain for distinguishing photographic images and computer graphics. *Multimedia Tools and Applications*, vol. 76, pp. 23721-23737.

Xu, D.; Wang, R. (2016): Separable and error-free reversible data hiding in encrypted images. *Signal Processing*, vol. 123, pp. 9-21

Yin, B.; Chen, F.; He, H.; Yan, S. (2017): Separable reversible data hiding in encrypted image with classification permutation. *IEEE Third International Conference on Multimedia Big Data*, pp. 201-204.

Yin, Z.; Abel, A.; Tang, J.; Zhang, X.; Luo, B. (2017): Reversible data hiding in encrypted images based on multilevel encryption and block histogram modification. *Multimedia Tools and Applications*, vol. 76, pp. 3899-3920.

Zhang, X. (2011): Reversible data hiding in encrypted image. *IEEE Signal Processing Letter*, vol. 18, no. 4, pp. 255-258.

Zhang, Y.; Qin, C.; Zhang, W.; Liu, F.; Luo, X. (2018): On the fault-tolerant performance for a class of robust image steganography. *Signal Processing*, vol. 146, pp. 99-111.

Zhou, J.; Sun, W.; Dong, L.; Liu, X.; Au, O. C. et al. (2016): Secure reversible image data hiding over encrypted domain via key modulation. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 3, pp. 441-452.