Multimedia Systems (2021) 27:303–315
https://doi.org/10.1007/s00530-020-00739-5

SPECIAL ISSUE PAPER

# Reversible data hiding in encrypted medical DICOM image

**Ping Kong[1,2] · Di Fu[2] · Xinran Li[4] · Chuan Qin[3,4]**

Received: 2 September 2020 / Accepted: 12 December 2020 / Published online: 4 January 2021
© The Author(s), under exclusive licence to Springer-Verlag GmbH, DE part of Springer Nature 2021

**Abstract**
This paper proposes a novel reversible data hiding (RDH) scheme in encrypted domain for medical DICOM images. Although a lot of RDH schemes for encrypted images have been presented, however, most of them are unsuitable for medical DICOM images, as they do not utilize the features of the DICOM image format, and the recovery accuracy is low because medical images have large areas with the same pixel values. To avoid these weaknesses, our scheme segments the image and only embeds data into part of the encrypted image. The redundancy of pixel cells in DICOM images is exploited so that auxiliary data can be embedded into the image. The order of data extraction and image recovery, which can be optimized by the number of recovered blocks around the unrecovered block, is calculated to improve the recovery accuracy. In addition, to ensure the integrity of the DICOM image, the hash value of the minimum bounding rectangle of ROI and the feature bit matrix of the rest are calculated. If the image is tampered, the tampered area, especially ROI, can be accurately detected and located on the receiver side. Experimental results demonstrate the effectiveness of the proposed scheme.

**Keywords** Reversible data hiding · DICOM image · Image encryption · Tampering detection

## 1 Introduction

As a technique for privacy protection, reversible data hiding (RDH) can embed secret messages into an image while the image is almost unchanged. In addition, this technique can reversibly recover the original image after performing data extraction on the receiver side. RDH is becoming increasingly more significant in the field of privacy security, especially in the medical and military fields where high image quality is required. People hope to embed more information into an image while ensuring the quality of the marked image. Embedding capacity and peak signal-to-noise ratio (PSNR) of the marked image are important criteria to evaluate the performance of schemes. The higher the PSNR is, the smaller the difference between the marked image and the original image. In the past two decades, a large number of RDH schemes have emerged. Examples include lossless compression [1] difference expansion [2], and histogram shifting [3].

### 1.1 Reversible data hiding

In the earlier stage of the research, messages were usually embedded in the plaintext domain. Fridrich et al. [1] suggested a lossless compression method to vacate room for additional information. The scheme in [2] expanded the pixel differences to provide space for secret message. Ni et al. [3] embedded secret message by shifting peak point to zero point. The scheme in [4] introduced a difference expansion (DE) mechanism using a pixel predictor for reversible data embedding.

✉ Chuan Qin
qin@usst.edu.cn

Ping Kong
kongp@sumhs.edu.cn

Di Fu
193832382@st.usst.edu.cn

Xinran Li
ranusst@163.com

1 Shanghai Key Laboratory of Molecular Imaging, Shanghai University of Medicine and Health Sciences, Shanghai 201318, China

2 School of Medical Instrument and Food Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China

3 School of Cyber Security, Qilu University of Technology, Jinan 250353, Shandong, China

4 School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China

### 1.2 Joint reversible data hiding in encrypted images

As cloud computing has developed, the combination of encryption and RDH has received much attention. To protect privacy, an image owner can encrypt multimedia files before their transmission. For better management, an administrator can embed auxiliary data, such as tags and provenance information, into encrypted images. Reversible data hiding in encrypted images (RDH-EI) is effective in a scenario where high image quality is required. There are considerable works on RDH-EI. Zhang [5] split an encrypted image and then partitioned each of the blocks into two parts, $S_0$ and $S_1$. For each block, the scheme flipped three LSBs of all pixels belonging to $S_0$ to embed slightly with a value of 0 and vice versa. Considering the correlations in natural images, the entropy of the block that was decrypted correctly was generally lower than that of a block that was decrypted incorrectly. In [6], Hong et al. improved Zhang's scheme. A better estimation method of block smoothness was presented. Moreover, decryption was performed from the block with the largest difference in its smoothness. In addition, the boundary pixels of the recovered blocks were considered by the side match strategy. The scheme [7] reduced the number of pixels with flipped LSBs. Furthermore, Hong et al. introduced a flipping ratio in [8]. In the above schemes, image recovery is tied to data extraction, and so the solution is non-separable.

### 1.3 Separable reversible data hiding in encrypted images

In contrast, Zhang [9] introduced a separable scheme. In [10], Wu et al. introduced a joint scheme and a separable scheme via the prediction error. In [11], a separable scheme based on pixel value ordering and additive homomorphism was presented. Qin [12] investigated a separable scheme via an adaptive embedding strategy with block selection. A separable scheme with a higher embedding rate based on an adaptive encoding strategy was introduced [13]. In [14], a separable scheme with a better distortion rate and redundancy transfer was proposed. Due to the focus on achieving a higher embedding capacity in an encrypted image, high-capacity RDH-EI schemes have been proposed [15, 16]. RDH-EI schemes with high embedding capacity based on MSB prediction were introduced in [17, 18].

However, mostly conventional schemes are designed for bitmap or JPEG images. Medical DICOM (digital imaging and communications in medicine) images usually have large areas with the same pixel values, and the recovery accuracy is low if using above schemes directly. To utilize the characteristics of DICOM images and ensure the recovery accuracy, in this work, we propose a novel reversible data hiding scheme for encrypted DICOM images. The main contributions of our scheme can be described as follows: (1) The order of data extraction and image recovery is calculated on the sender side, and inspired by the side match mechanism, the order of extraction and recovery is further optimized during the recovery process on the receiver side, and the to-be-recovered block that has greater number of neighbouring, recovered blocks results in higher recovery priority; (2) A DICOM image is segmented into the ROI (region of interest) and the RONI (region of non-interest), and only the minimum bounding rectangle of ROI is embedded with additional data after encryption, which can guarantee the recovery accuracy effectively; (3) To ensure image integrity, the hash value of the minimum bounding rectangle of ROI and the feature bit matrix of the rest region are calculated, respectively; (4) The redundancy in pixel cells of DICOM image is exploited to hide auxiliary data without destroying the image format.

The remainder of this paper is arranged as follows. Section 2 introduces the proposed scheme in detail. Experimental results and analyses are given in Section 3. Conclusions are drawn in Section 4.

## 2 Proposed scheme

The proposed scheme consists of five stages: pre-processing, image encryption, data embedding, data extraction and image recovery, and tampering detection, respectively, see Fig. 1. In the first stage, the auxiliary data, including order of extraction and recovery, hash value, and feature bit matrix, are calculated to replace the redundancy of pixel cells in the DICOM image. At the stage of image encryption, an encrypted DICOM image is generated by the encryption key. In the third stage, with the data hiding key, the data hider embeds the encrypted personal information into the encrypted DICOM image. After decryption on the receiver side, the marked DICOM image can be obtained, and the embedded data are then extracted while the original image can be recovered. At last, the receiver recalculates the hash value and feature bit matrix. If the calculated results are consistent with the extracted results, it means that the image has not been tampered. Otherwise, the image is judged as tampered and the area where the feature values are different can be located. The main parameters used in the experiments are listed in Table 1.

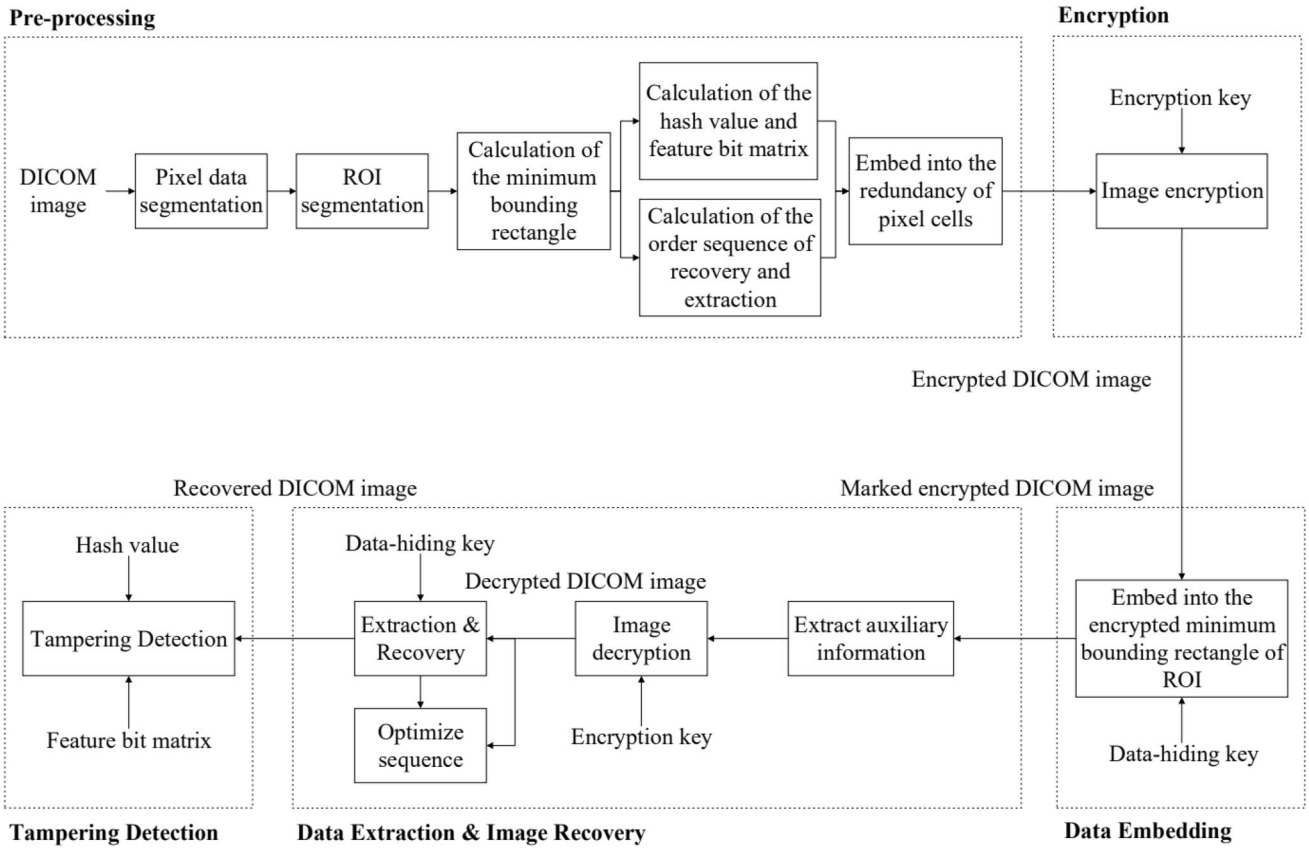


**Fig. 1** Flowchart of the proposed scheme

**Table 1** Parameters in the proposed scheme

| Parameters | Descriptions |
|---|---|
| $s$ | The side length of the block |
| $M$ | The number of flipped bit planes |
| $O$ | The length of the order sequence fragment |
| $D$ | The table recording the number of recovered blocks around each block |

## 2.1 Pre-processing

In this step, the original image is first segmented into the minimum bounding rectangle of ROI (i.e., $R_1$) and the rest (i.e., $R_2$). The minimum bounding rectangle is an expression of the maximum extents of a 2-dimensional object. As shown in Fig. 2, the minimum bounding rectangle of ROI is outlined in blue. Then the auxiliary data are calculated

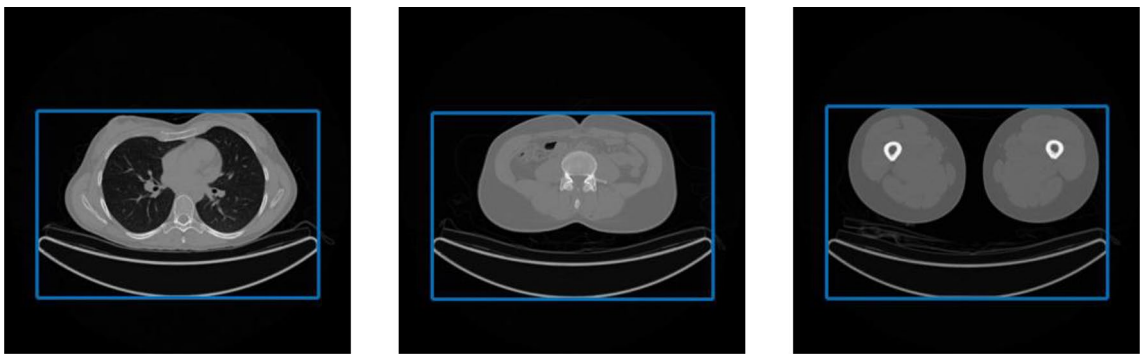

**Fig. 2** The minimum bounding rectangle of ROI

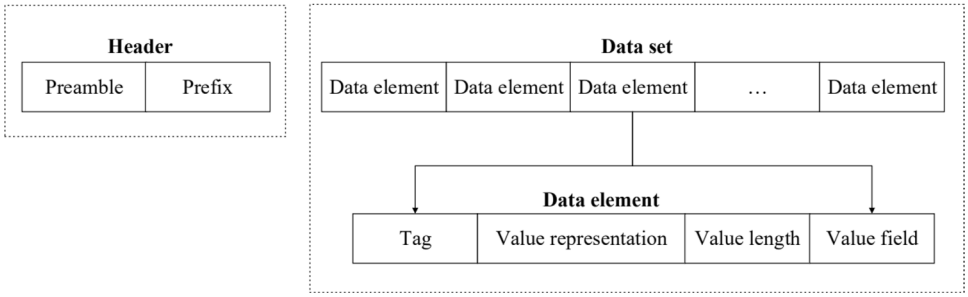


**Fig. 3** The structure of the medical DICOM image file

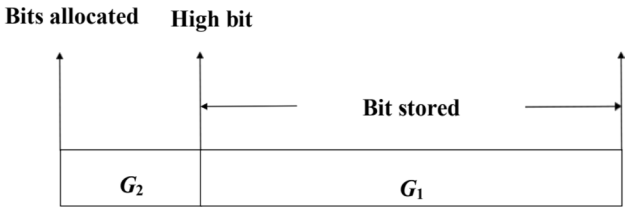


**Fig. 4** The structure of a pixel cell

and embedded into the redundancy of pixel cells in the DICOM file. A DICOM file consists of a header and a data set [19]. As shown in Fig. 3, the basic unit of the data set is a data element. The personal information (patient name, admitting diagnosis description, additional patient history, etc.), pixel data, and the image parameters (bits stored, high bit, bits allocated, etc.) are stored in the data elements. The data element can be recognized by its tag. For example, (0010, 0010) is the tag of data element named patient name, and (7FE0, 0010) is the tag of data element named pixel data. In addition, pixel data are composed of pixel cells. It is worth mentioning that the space of a pixel cell is not always fully utilized. How the pixel value is stored in the pixel sample is illustrated in Fig. 4. For example, the number of allocated bits in CT images is generally 16 while only 12 bits in $G_1$ are used for pixel sampling and the remaining 4 bits in $G_2$ are redundant. To obtain the bits in $G_1$, the bits allocated, the high bit, and the bits stored in data elements must be recognized first. Then, the pixel matrix that consists of $G_1$ can be obtained.

There are regions with the same pixel values in medical images, which may cause the recovery based on image correlation has a high error rate. Therefore, in our scheme, image segmentation is applied to images to avoid this weakness. In this process, Otsu's method is adopted and an image is segmented into the ROI and RONI. Next, calculate the minimum bounding rectangle of ROI and record the top left and bottom right vertices. Replace the bits in $G_2$ of the last 10 pixel cells with the bits of the vertices. Then, divide the minimum bounding rectangle of ROI (i.e., $R_1$) into several non-overlapping blocks sized $s \times s$. Denote the block located at $(i, j)$ as $H^{i,j}$. The smoothness index $\sigma$ of the block is calculated by Eqs. (1), (2):

$$\overline{X} = \frac{\sum_{u=1}^{s}\sum_{v=1}^{s} p_{u,v}}{s \times s}, \tag{1}$$

$$\sigma = \frac{\sum_{u=1}^{s}\sum_{v=1}^{s} \left(p_{u,v} - \overline{X}\right)^2}{s \times s}. \tag{2}$$

where $p_{u,v}$ indicates the pixel at the coordinate $(u, v)$ in the block. According to the observation, the recovery accuracy of a smooth block is higher than that of a complex block. The larger the index $\sigma$ is, the more complex the block is. The blocks are sorted according to their smoothness indices $\sigma$ in the ascending order. Denote the order sequence of sorted block order as $L$, where $L$ is composed of coordinates of all blocks.

To detect whether and where the image has been tampered, the hash value of region $R_1$ and the feature bit matrix of region $R_2$ are calculated by different methods. For region $R_2$, we divide the area into several non-overlapping blocks sized $4 \times 4$. Denote the blocks as $T_1^{1,1}, T_1^{1,2}, \ldots, T_1^{i',j'}, \ldots$, where $(i', j')$ represents the block location. A two-level lifting wavelet transform is applied on each block $T_1^{i',j'}$ and a low-frequency approximation sub-band $C_{i',j'}$ sized $4 \times 4$ can be obtained by Eq. (3). A new image that is one-sixteenth the size of the original rectangle is generated by combining the sub-bands. Furthermore, generate a feature bit matrix $\tau_1$ from the 5th bit plane of the new image using Eq. (4).

$$C_{i',j'} = \mathrm{lwt2}\left(T_1^{i',j'}\right), \quad T_1^{i',j'} \in R_2, \tag{3}$$

$$\tau_1^{i',j'} = \left\lfloor \frac{C_{i',j'}}{2^5} \right\rfloor \bmod 2. \tag{4}$$

For region $R_1$, to consider higher importance, we use the cryptographic Hash function, MD5, to calculate the hash value $\tau_2^{i'',j''}$ of each pixel $T_2^{i'',j''}$ in region $R_1$ to achieve better

performance of tampering detection, where $(i'', j'')$ indicates the pixel location. The hash value $\tau_2^{i'',j''}$ is calculated using Eq. (5).

$$\tau_2^{i'',j''} = \mathrm{MD5}(T_2^{i'',j''}), \quad T_2^{i'',j''} \in R_1. \tag{5}$$

Finally, replace the bits in $G_2$ with the order sequence $L$, the feature bit matrix $\tau_1$, and the hash value $\tau_2$ to finish the pre-processing stage. A summary of the pre-processing stage is listed in Algorithm 1.

**Algorithm 1** Pre-processing

**Input**: DICOM image

1: Read the data elements' pixel data, personal information, bits allocated, high bit, and bits stored according to the tags.
2: **for** each pixel in pixel data **do**
3: According to the allocated bits, high bit, and bits stored, classify the pixel cell into $G_1$ and $G_2$.
4: Calculate the grey value according to the bits in $G_1$.
5: **end for**
6: Segment the image into two regions $R_1$ and $R_2$.
7: Divide region $R_1$ into equal sized blocks.
8: **for** each block in region $R_1$ **do**
9: Calculate the block smoothness index $\sigma$.
10: **end for**
11: Sort the blocks by smoothness index $\sigma$ from small to large and record the order sequence $L$.
12: Divide region $R_2$ into several non-overlapping blocks sized $4 \times 4$.
13: **for** each block $T_1^{i',j'}$ in region $R_2$ **do**
14: Calculate two-level lifting wavelet transform.
15: Obtain the low frequency sub-band $C_{i',j'}$.
16: Generate the feature bit $\tau_1{}^{i',j'}$.
17: **end for**
18: **for** each pixel $T_2^{i'',j''}$ in region $R_1$ **do**
19: Calculate the MD5 value $\tau_2{}^{i'',j''}$.
20: **end for**
21: **for** each pixel cell in the pixel data **do**
22: **for** each bit in $G_2$ **do**
23: Replace the original bit with the bit in the order sequence $L$, and feature bit matrix $\tau_1$, hash value $\tau_2$.
24: **end for**
25: **end for**

**Output**: Pre-processed DICOM image

## 2.2 Image encryption

For a DICOM image, denote the bits in a pixel cell $P_{u,\,v}$ containing $G_1$ and $G_2$ as $b_{u,v,0}, b_{u,v,1}, \ldots, b_{u,v,15}$. Calculate the exclusive-or (XOR) results of the bits in the pixel cells and pseudo-random bit $r_{u,v,k}$.

$$B_{u,v,k} = b_{u,v,k} \oplus r_{u,v,k}. \tag{6}$$

Encrypt the value field of the personal information data elements in the same way. Actually, in addition to stream cipher methods, block cipher methods, such as DES and AES, can also be utilized.

## 2.3 Data embedding

To perform data embedding, the encrypted personal information, the encrypted pixel data, bits stored, high bit, and bits allocated data elements should be recognized first. Then, classify all pixel cells in the pixel data into $G_1$ and $G_2$. The encrypted personal information and the auxiliary information that can be embedded into the encrypted pixel matrix consist of $G_1$. Next, read the vertices from the last 10 pixel cells, and $R_1$ can be obtained. Then, segment region $R_1$ into several non-overlapping blocks sized $s \times s$. Each of these blocks is pseudo-randomly partitioned into $S_0$ and $S_1$ by the data-hiding key. Finally, flip the $M$ LSBs of each encrypted pixel belonging to $S_0$ or $S_1$ to embed 0 or 1 correspondingly, see Eqs. (7), (8).

$$B'_{u,v,k} = \overline{B_{u,v,k}}, \quad (u,\ v) \in S_0 \text{ and } k = 0,\ 1,\ \ldots,\ M, \tag{7}$$

$$B'_{u,v,k} = \overline{B_{u,v,k}}, \quad (u,\ v) \in S_1 \text{ and } k = 0,\ 1,\ \ldots,\ M. \tag{8}$$

## 2.4 Data extraction and image recovery

To decrypt and recover a marked encrypted DICOM image, the receiver has to recognize the pixel data, bits stored, high bit, and bits allocated data elements first. Then, divide the pixel cells into $G_1$ and $G_2$ after decryption. The order sequence $L$, feature bit matrix $\tau_1$, and hash value $\tau_2$ are extracted from $G_2$. Next, read the vertices from the last 10 pixel cells and the region $R_1$ is obtained. Then, divide the decrypted region $R_1$ that consists of $G_1$ into several equal blocks sized $s \times s$, i.e., $H'^{1,2}$, $H'^{1,2}$, …, $H'^{i,j}$, where the block located at $(i, j)$ is denoted as $H'^{i,j}$. Using the data-hiding key, each block is split into $S_0$ and $S_1$. Next, flip the LSBs in set $S_0$ and $S_1$ to produce $H_0^{i,j}$ and $H_1^{i,j}$, respectively. Finally, evaluate the fluctuation of $H_0^{i,j}$ and $H_1^{i,j}$ with Eq. (9):

$$f = \sum_{u=1}^{s}\sum_{v=1}^{s-1}|p_{u,v} - p_{u,v+1}| + \sum_{u=1}^{s-1}\sum_{v=1}^{s}|p_{u,v} - p_{u+1,v}|. \quad (9)$$

Denote the evaluated fluctuation results for $H_0^{i,j}$ and $H_1^{i,j}$ as $f_0^{i,j}$ and $f_1^{i,j}$, respectively. Because the natural images have spatial correlation, original block usually has smaller fluctuations. Hence, if $f_0^{i,j} < f_1^{i,j}$, let $H_0^{i,j}$ be the original block and the extracted bit is 0; otherwise, let $H_1^{i,j}$ be the original block and the extracted bit is 1. As mentioned in the preprocessing, the lower the image complexity is, the higher the accuracy of the recovery is. Therefore, the blocks are recovered in the order based on the extracted order sequence $L$. In addition, the order sequence $L$ can be further optimized during the extraction and recovery process. For example, Hong et al. introduced a side match mechanism to further increase recovery accuracy [6]. For each recovered block, the boundary pixels are concatenated to the blocks not recovered around it. Because of the high correlation, the recovery accuracy of the concatenated blocks can be higher.

Inspired by the side match mechanism, our scheme further optimizes the order sequence $L$ by dividing the unrecovered blocks into five categories: 0–4. These categories are determined by the number of recovered blocks around them. For a block to be recovered, the more recovered blocks there are, the more pixels concatenated to the unrecovered block are. Build a table $D$ to record the number of recovered blocks around each block. At the beginning of recovery, all the numbers in $D$ are 0. As the recovery proceeds with the order sequence $L$, the number in $D$ is also updated. Segment the order sequence $L$ into several small fragments sized $O$, and for each small fragment, the blocks in it have similar smoothness while the blocks in $L$ are sorted by the smoothness. Reorder the fragment in descending order according to the numbers in $D$. Then, follow the steps above for the blocks in the next fragment until the blocks in all fragments are recovered.

### 2.5 Tampering detection

After the extraction of the feature bit matrix $\tau_1$, and hash value $\tau_2$ from the redundancy part $G_2$ of pixel data, partition the image into two regions $R_1$ and $R_2$ using the vertices from the last 10 pixel cells. Next, recalculate the feature bit matrix of the recovered region $R_2$ and the hash value of the recovered region $R_1$. Then, compare the recalculated hash value and feature bit matrix with the extracted hash value and feature bit matrix. The image can be judged as intact if the values are exactly the same. Otherwise, the image area where the values are different can be detected as tampered. Note that, the tampering localization for $R_1$ is more accurate than that for $R_2$, since the scheme calculates the hash value for each pixel in $R_1$ and the feature bit matrix for blocks sized $4\times4$ in $R_2$.

## 3 Experimental results and analyses

To demonstrate the performance of the proposed scheme, we conducted experiments on a large number of medical DICOM images [20]. Four computed tomography (CT) DICOM images are shown in Fig. 5a–d and the pixel value range is between 0 and 4095, i.e., the bit depth is 12.

### 3.1 Result of our scheme

The test was conducted using the parameters of $s = 16$, $M = 5$, and $O = 128$. Figure 6a is a binary image for Fig. 5a generated by Otsu's method, which labels ROI and RONI. It can be seen from the figure that Otsu's method is suitable for medical images. Figure 6b is the minimum bounding rectangle of the ROI in Fig. 5a. Figure 6c is the encrypted image for Fig. 5a. After the data are embedded into Fig. 5c, the marked, encrypted image, Fig. 6d, is generated. Figure 6e is the directly-decrypted image. After data extraction and image recovery, the recovered image can be generated reversibly, see Fig. 6f.

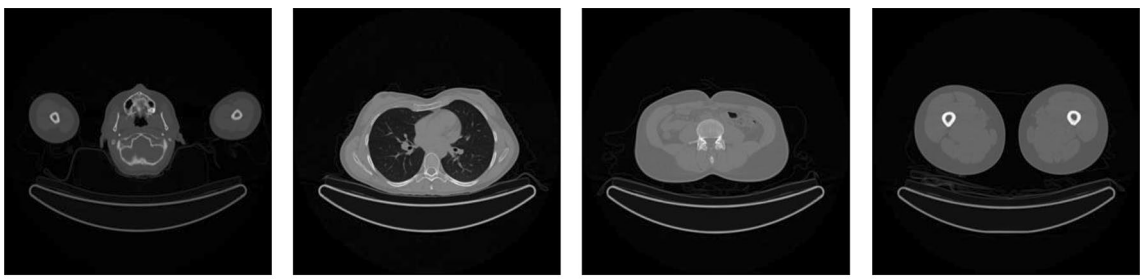


**Fig. 5** Four test DICOM images

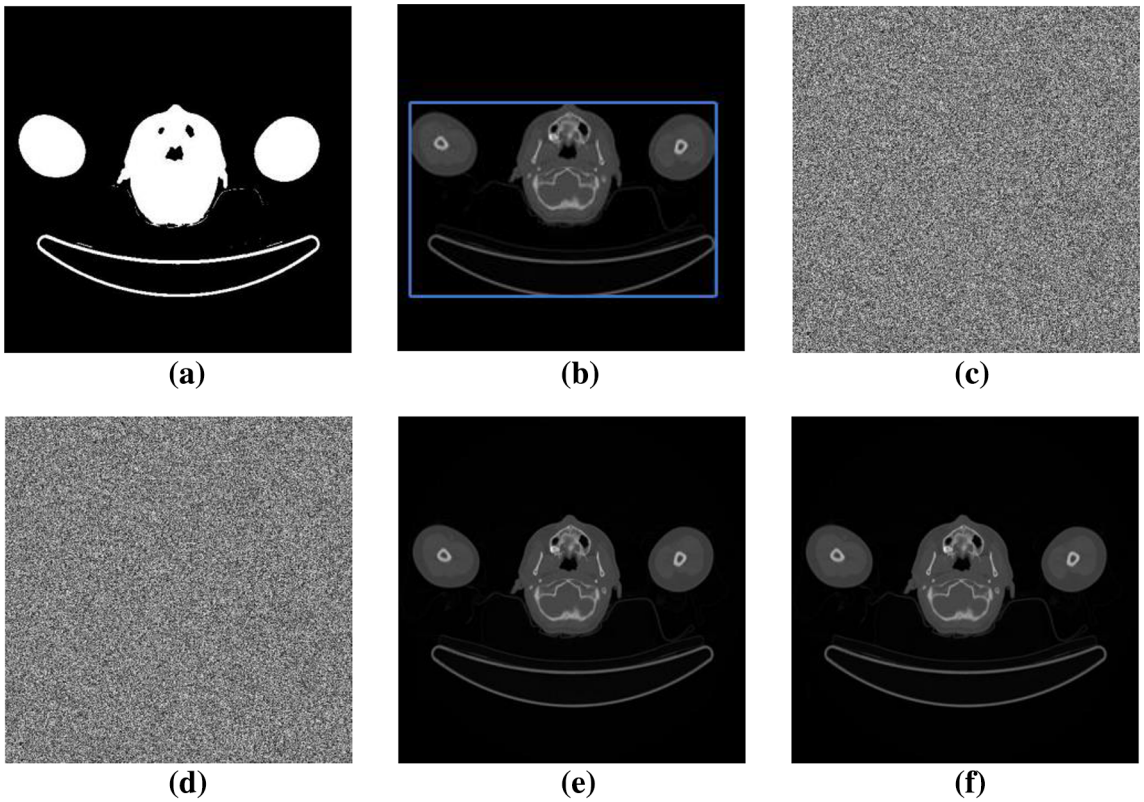


**Fig. 6** An example of our scheme. **a** Binary Image labelling ROI and RONI, **b** minimum bounding rectangle, **c** encrypted image, **d** marked, encrypted image, **e** decrypted image, **f** recovered image

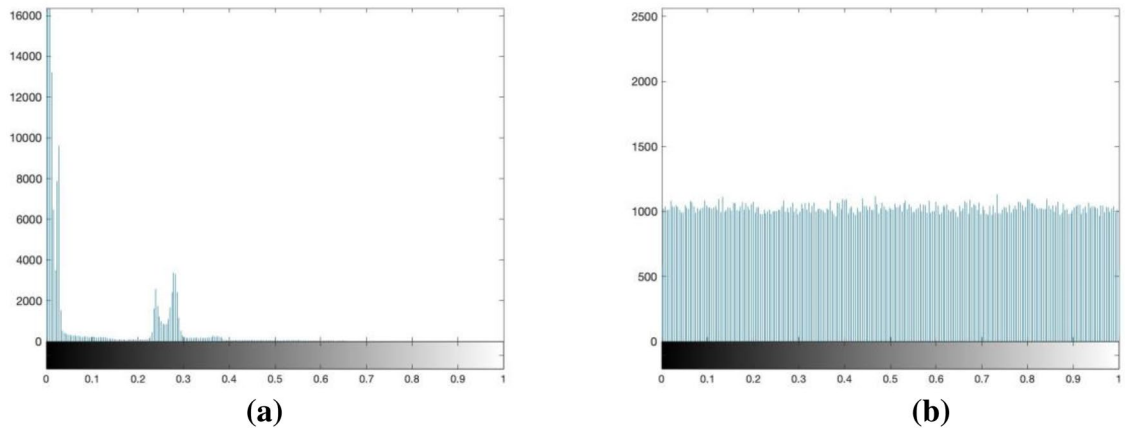


**Fig. 7** **a** Histogram of the original DICOM image in Fig. 5a. **b** Histogram of and the encrypted DICOM image in Fig. 6c

**Table 2** Order-1 entropy and pixel correlation

| Images | Pixel correlation | | | Order-1 entropy (bits) |
|---|---|---|---|---|
| | Horizontal | Vertical | Diagonal | |
| Original image | 0.9905 | 0.9877 | 0.9786 | 6.7207 |
| Encrypted image | −0.0049 | 0.0232 | −0.0140 | 11.9884 |

## 3.2 Security analysis

To verify the security of the proposed scheme, we also conducted experiments on the original image in Fig. 5a and its encrypted version in Fig. 6c. Figure 7a shows the histogram of the original image, and the pixels are mostly on the left. Figure 7b shows the corresponding encrypted image, and

the pixels are uniformly distributed. Therefore, the proposed scheme is secure regarding its statistical characteristics.

Furthermore, the pixel correlation is listed in Table 2. The correlation values of the original image are close to 1 while those of the encrypted image are close to 0. In addition, Table 2 shows the information entropy. The system with greater information entropy is more random. The Order-1 entropies $\mu$ are calculated with Eq. (10):

$$\mu = -\sum_{i=0}^{4095} p_i \log_2 p_i, \tag{10}$$

where $p_i$ represents the proportion of pixels with the gray value $i$ in the image.

For a more intuitive display of results, Fig. 8a–f illustrate the correlation distributions. Obviously, the pixel correlation in the original image is stronger while the encrypted image has a weaker pixel correlation.

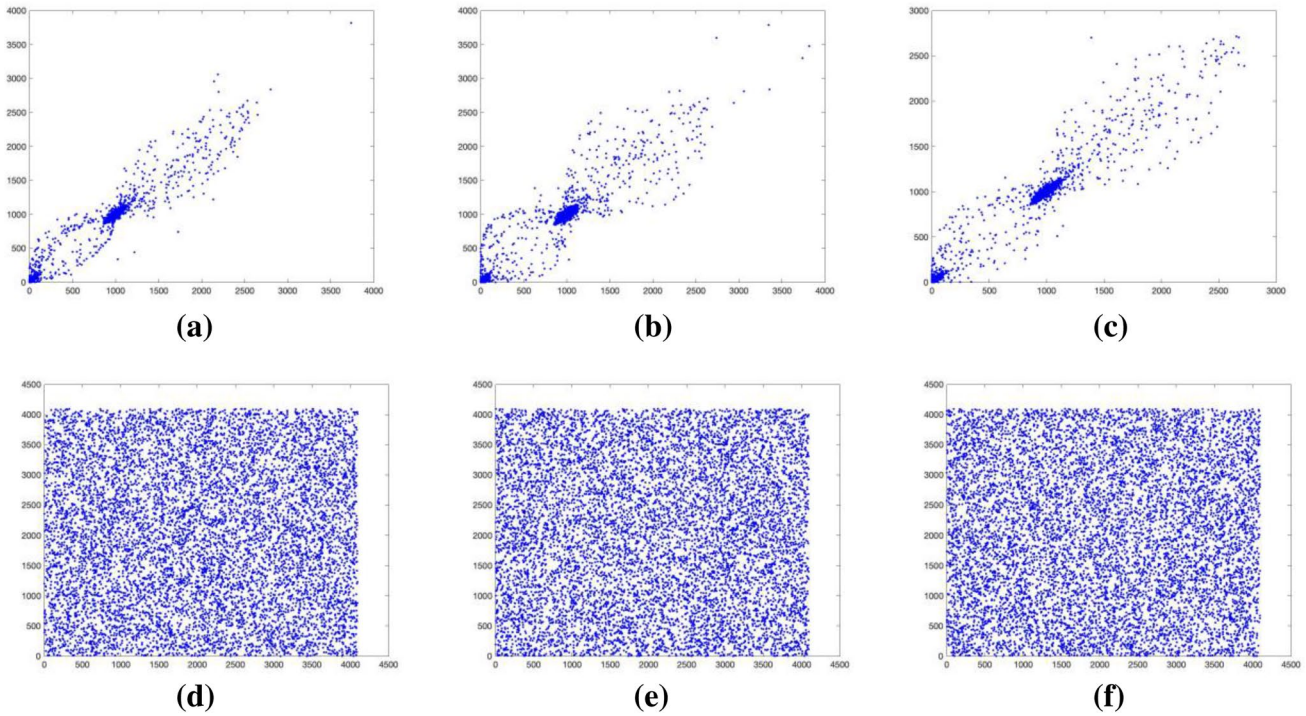


**Fig. 8** Pixel correlations of original DICOM image in Fig. 5a and the corresponding encrypted image in Fig. 6c. **a** Horizontal direction of the original image, **b** vertical direction of the original image, **c** diagonal direction of the original image, **d** horizontal direction of the encrypted image, **e** vertical direction of the encrypted image, and **f** diagonal direction of the encrypted image

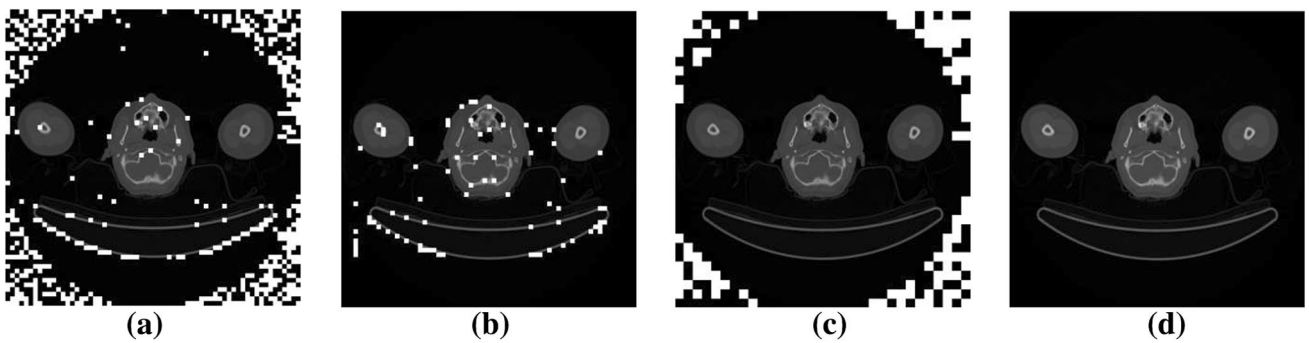


**Fig. 9** Comparisons of recovery performance. **a** Recovered BMP image by Hong et al. scheme [6] under $s=8$, **b** recovered DICOM image by the proposed scheme under $s=8$, **c** recovered BMP image by Hong et al. scheme [6] under $s=16$, and **d** recovered DICOM image by the proposed scheme under $s=16$

### 3.3 Recovery performance

To demonstrate the superiority of the proposed scheme on the recovery accuracy, we compare the recovery accuracy of the DICOM image recovered by the proposed scheme with that of the BMP image converted from DICOM format by Hong et al. scheme [6], under $s=8$ in Fig. 9a, b and $s=16$ in Fig. 9c, d, respectively. It is noteworthy that the recovery accuracy is low at the edge of the image recovered by Hong et al. scheme [6]. Because the pixel values of image four corners are exactly the same and $f_0^{i,j}$ and $f_1^{i,j}$ are consequently exactly the same, it is hard to recover these areas. In the proposed scheme, these areas are excluded and only the minimum bounding rectangle of ROI is embedded with data. As a result, the recovery accuracy of our scheme is improved significantly. Figure 10a–d show the experimental results on Fig. 5a–d with different block side lengths $s$, and the results imply that the proposed scheme performs better than [6] with respect to the recovery accuracy. It should be noted that, the recovery accuracy is calculated upon the size of the minimum bounding rectangle of ROI because the rest of the image is not embedded.

**Table 3** Average extracted-bits error rate under different parameters

| | $s$ | $M$ | Average error rate (%) | |
|---|---|---|---|---|
| | | | Proposed scheme | Hong et al.'s scheme [6] |
| 500 DICOM images [20] | 8 | 3 | 7.7 | 14.69 |
| | 12 | 3 | 3.67 | 10.94 |
| | 16 | 3 | 1.45 | 9.89 |
| | 20 | 3 | 0.49 | 7.87 |
| | 8 | 5 | 6.29 | 17.34 |
| | 12 | 5 | 2.49 | 12.15 |
| | 16 | 5 | 1.51 | 11.79 |
| | 20 | 5 | 0.81 | 8.66 |

In addition, experiments were conducted for 500 DICOM images [20] under different parameters $s$ and $M$ for

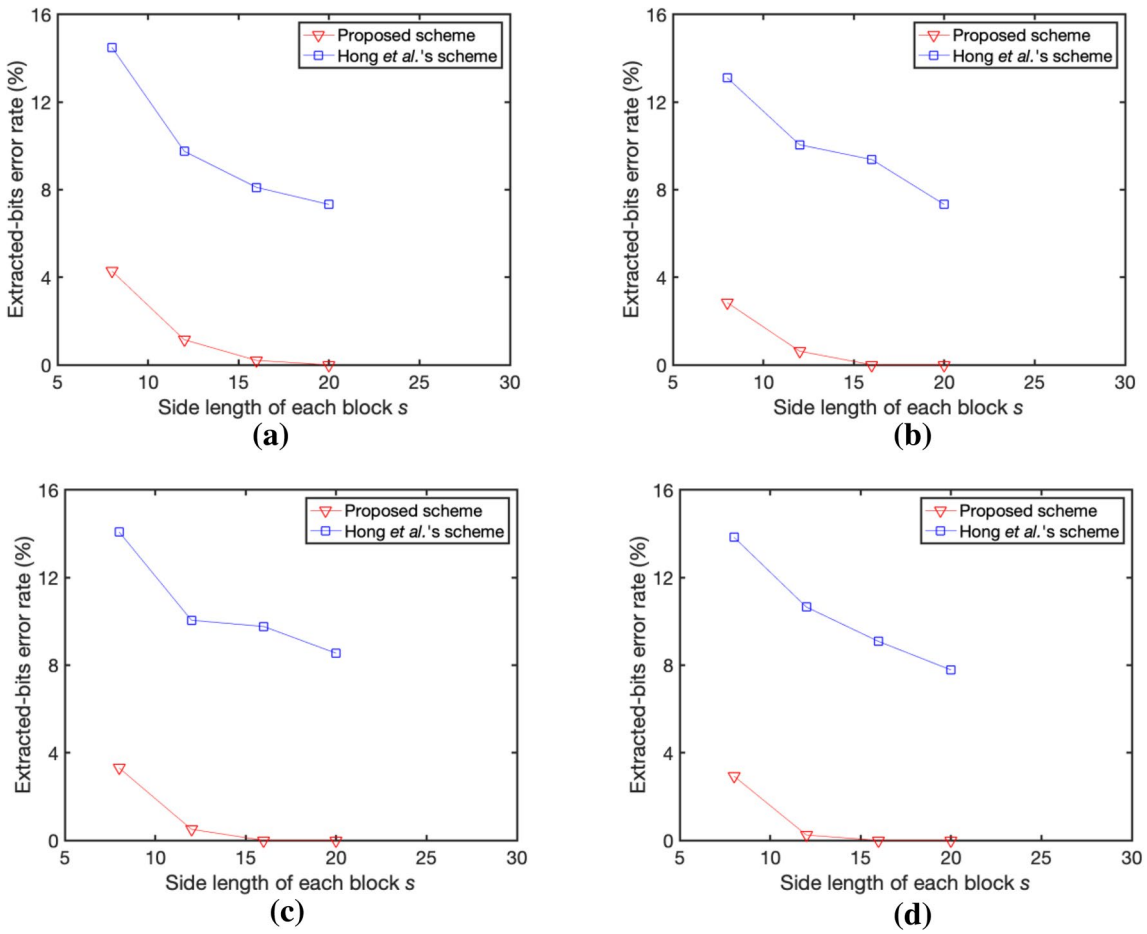


**Fig. 10** Comparisons of the extracted-bits error rate under $M=5$. **a** Extracted-bits error rate for Fig. 5a. **b** Extracted-bits error rate for Fig. 5b. **c** Extracted-bits error rate for Fig. 5c. **d** Extracted-bits error rate for Fig. 5d

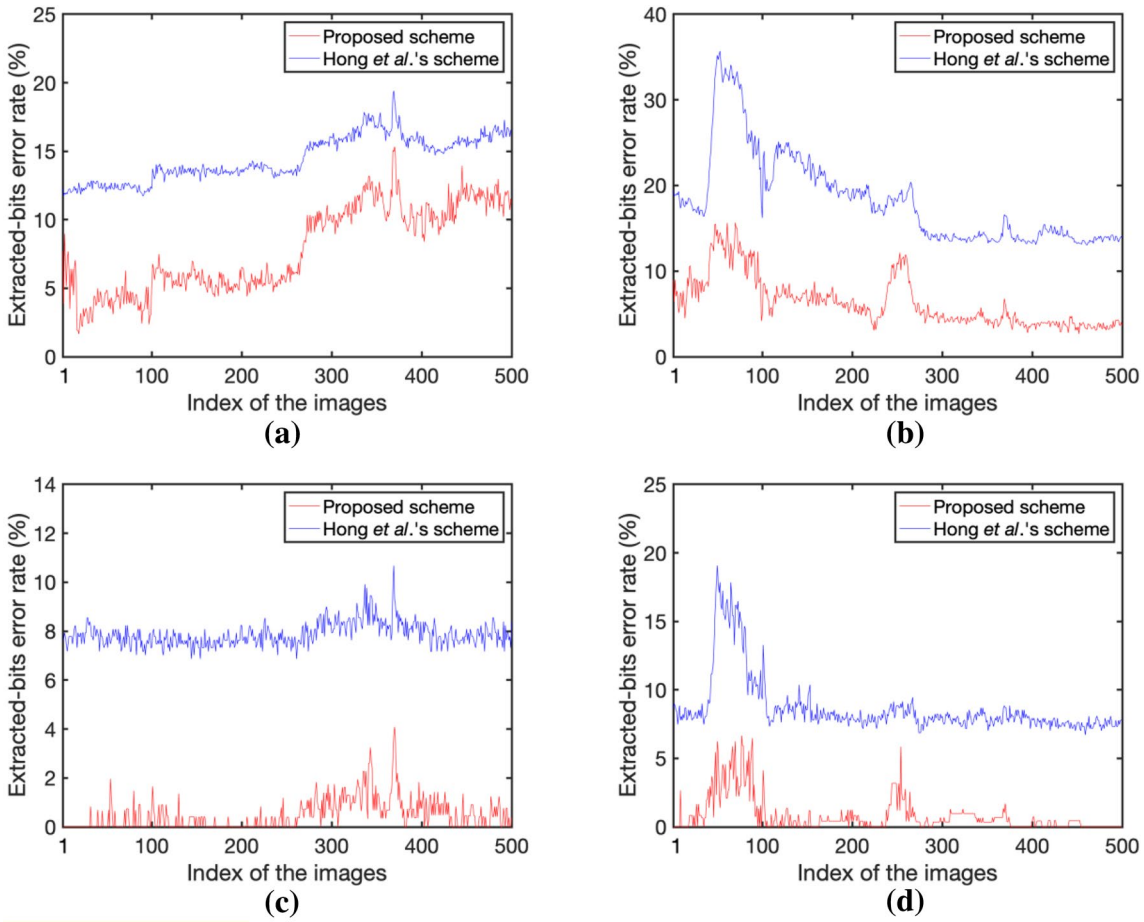


**Fig. 11** Extracted-bits error rate with different $M$ and $s$ ($O=128$). **a** $M=3$ and $s=8$, **b** $M=5$ and $s=8$, **c** $M=3$ and $s=20$, and **d** $M=5$ and $s=20$

performance analysis. Table 3 shows the results of using different values of $s$ and $M$ and that the average extracted-bits error rate of the proposed scheme is lower than that of [6].

For a more intuitive representation, we show the recovery accuracy of each image in the image database [20] under the parameters of $s=8$, $M=3$; $s=8$, $M=5$; $s=20$, $M=3$; $s=20$, $M=5$, respectively. The results in Fig. 11 show that the extracted-bits error rate of our scheme is much lower than that of Hong et al. scheme [6]. Since the image database [20] consists of continuous CT images, the features of image content are also different. In the image database [20], the pixel values of $R_1$ in images $1\sim280$ are small, and flipping 3 LSBs can effectively affect the image correlation. On the contrary, the pixel values of $R_1$ in images $281\sim500$ are large and flipping 3 LSBs can hardly affect the image correlation. Hence, the recovery effect when $M=3$ is better for the images $1\sim280$ when $M=5$ is better for the latter.

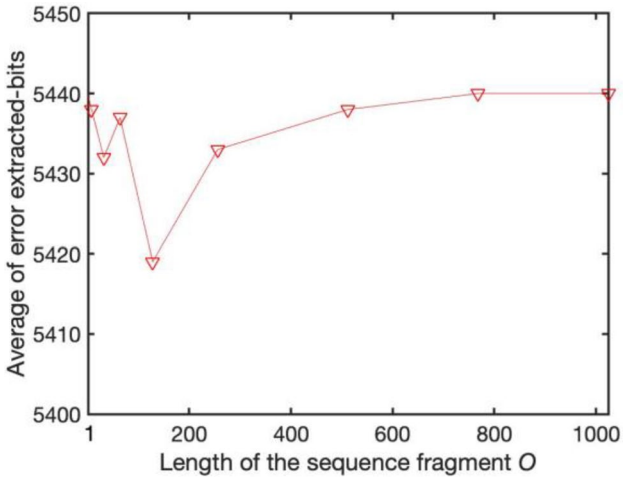


**Fig. 12** Average of error extracted-bits for 500 images in [20]

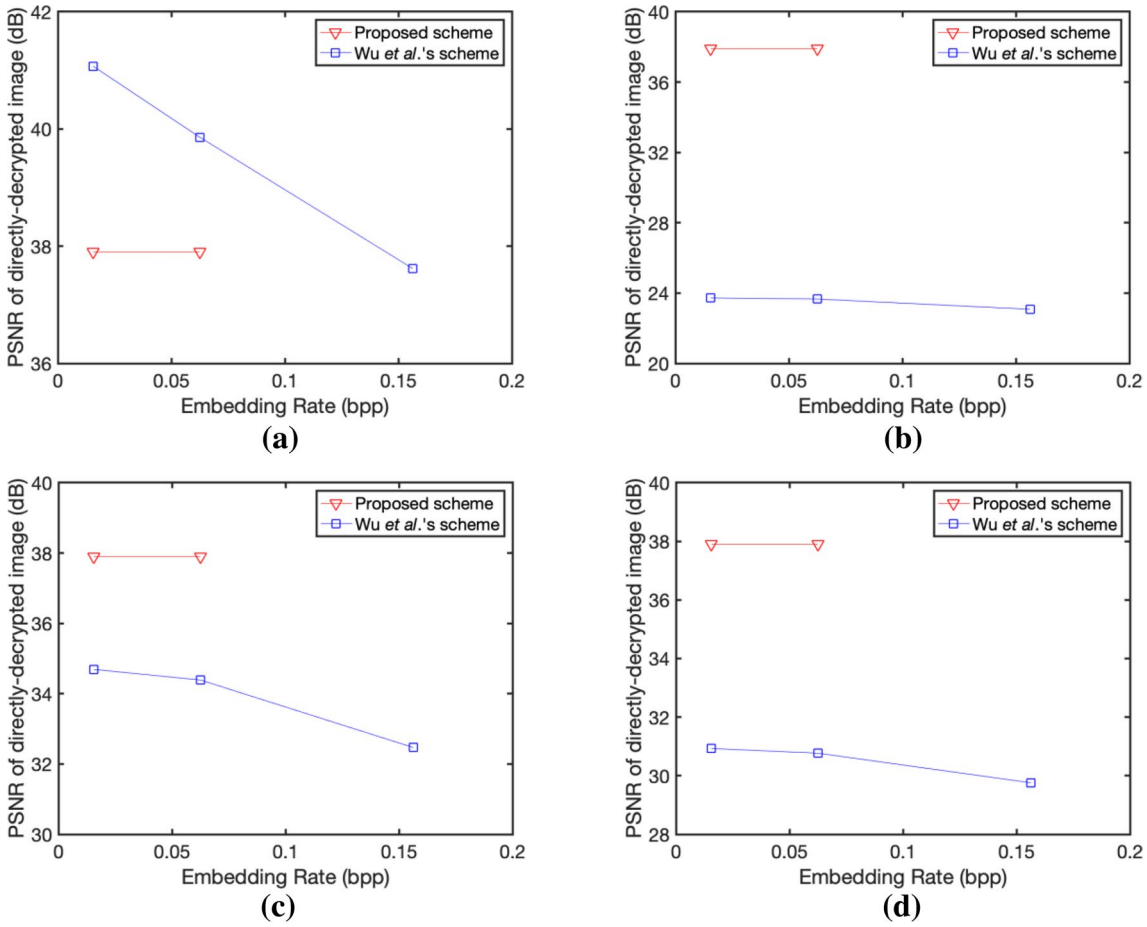


**Fig. 13** Comparisons for visual quality of directly-decrypted images between our scheme and [10]. **a** Lena, **b** baboon, **c** airplane, **d** lake

As $s$ increases, the extracted-bits error rate of both schemes decreases obviously. When $s=20$ and $M=5$, the extracted-bits error rate of the images with high pixel values can be zero.

The accuracy of data extraction and image recovery under different values of $O$ are shown in Fig. 12 ($s=16$, $M=5$). When $O$ is equal to 1, it means that the order sequence of data extraction and image recovery is not optimized. When the value of $O$ is too small, the order sequence fragments are almost unchanged and the recovery accuracy is almost unchanged. When the value of $O$ is too large, a complex block may be recovered earlier than a smooth block, and the recovery accuracy will not increase. Experiments on images of the image database [20] show that, when $O$ is equal to 128, the accuracy of data extraction and image recovery is better.

The visual quality of directly-decrypted images was also compared. Since this performance of our scheme is similar to that of Hong et al. scheme [6], we compared the visual quality of directly-decrypted image between our scheme and the scheme [10] in Fig. 13a–d. The scheme [10] adopted BMP images, Lena, Baboon, Airplane, and Lake, as original images, while the proposed scheme adopted the corresponding DICOM versions converted from the BMP images as original images. The results show that the visual quality of directly-decrypted images using the proposed scheme is stable. The PSNR of the proposed scheme is maintained at 37 dB while the scheme [10] is unstable, and the PSNR is below 37 dB in many cases.

### 3.4 Tampering detection performance

In Fig. 14a, only region $R_2$ was tampered by a text insertion attack. Conversely, region $R_1$ was tampered by a text insertion attack in Fig. 14b. In Fig. 14c, the image was tampered by a content removal attack. Figure 14d shows the result of tampering localization using a two-level lifting wavelet transform for the region $R_2$ in Fig. 14a. Figure 14e shows the result of tampering localization using MD5 Hash function for region $R_1$ in Fig. 14b. Comparing Fig. 14d and e, the tampering localization for region $R_1$ is obviously more accurate than that for region $R_2$, because

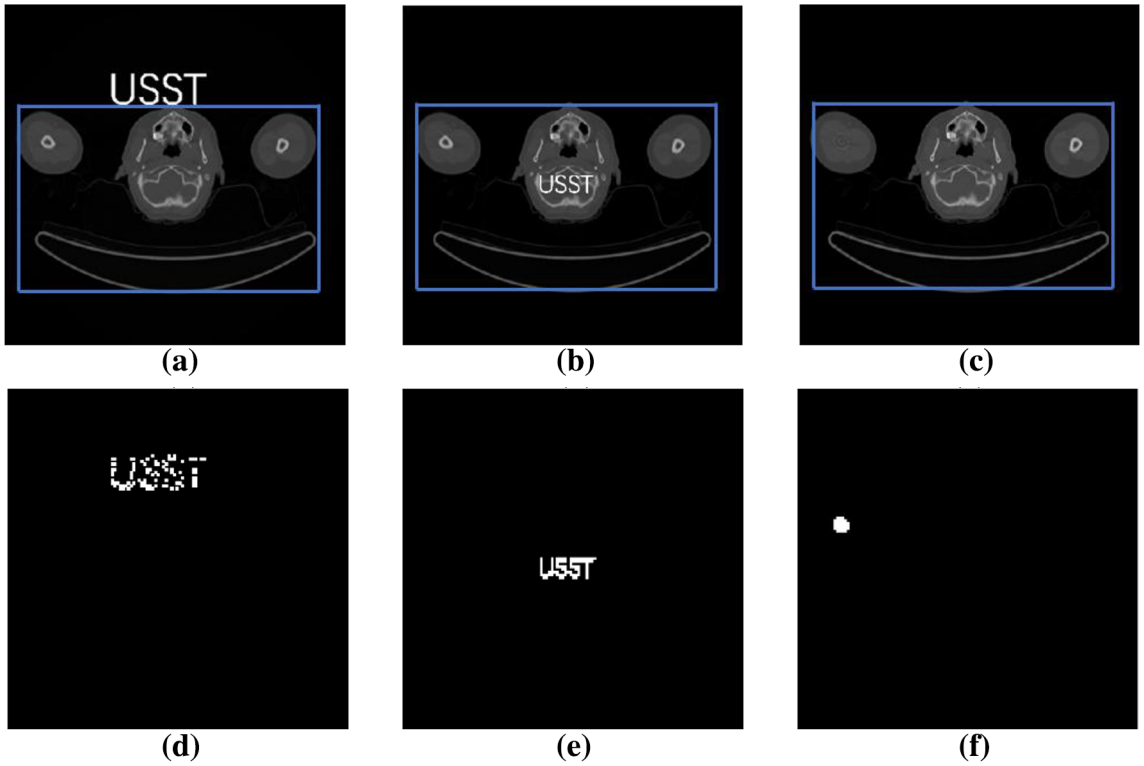


**Fig. 14** Results for tampering detection. **a** The region $R_2$ tampered by a text insertion attack, **b** the region $R_1$ tampered by a text insertion attack, **c** the region $R_1$ tampered by a content removal attack, **d** tampering localization for (**a**), **e** tampering localization for (**b**), and **f** tampering localization for (**c**)

**Table 4** Execution Time of each stage in the proposed scheme (seconds)

| Images | Pre-processing | Encryption | Embedding | Decryption | Recovery | Tampering detection |
|---|---|---|---|---|---|---|
| Figure 5a | 3.895 | 0.517 | 0.046 | 0.586 | 0.099 | 3.766 |
| Figure 5b | 4.176 | 0.554 | 0.079 | 0.593 | 0.132 | 3.785 |
| Figure 5c | 3.916 | 0.516 | 0.053 | 0.589 | 0.116 | 3.784 |
| Figure 5d | 4.809 | 0.527 | 0.045 | 0.585 | 0.118 | 4.302 |
| Image database [20] | 3.939 | 0.5185 | 0.040 | 0.594 | 0.098 | 3.883 |

we calculate the feature bit matrix of each block sized $4 \times 4$ in region $R_2$, while we calculate the hash value of each pixel in region $R_1$. Figure 14f shows the result of tampering localization for Fig. 14c. The area where contents haven been tampered is correctly located.

At last, we calculated the execution time of each stage for the proposed scheme, as listed in Table 4. The result shows that, the time spent on the receiver side is similar to that on the content owner side, and the time spent on the data hider sider is much less. Besides, most time spent on the two sides of receiver and content owner is utilized for the calculation of feature values and tampering detection.

## 4 Conclusion

This work proposes a novel RDH-EI scheme for medical DICOM images. DICOM images generally have large areas with the same pixel values, leading to low recovery accuracy with conventional RDH-EI schemes. To avoid this defect, the proposed scheme in this paper segmented the DICOM image into the ROI and RONI. Additional data are only embedded in the minimum bounding rectangle of the ROI where the data hiding can also be reversible. To further improve the recovery accuracy on the receiver side, a new order sequence of data extraction and image

recovery is calculated on the sender side. In addition, the order sequence can be optimized during the recovery process using the number of blocks around the to-be-recovered block. Based on the side match mechanism, the border pixels that belong to the recovered blocks are concatenated to the unrecovered blocks. Therefore, while the blocks have similar smoothness, the block with more recovered blocks around it has higher recovery accuracy. To ensure the integrity of the image, different tampering detection methods are used for the two types of regions in the image. Considering the importance of the ROI in medical images, the MD5 hash function is applied to the minimum bounding rectangle of ROI to locate the tampered area more accurately, and a two-level lifting wavelet transform is applied to the rest of the image. The redundancy of pixel cells in DICOM images is exploited so that the auxiliary data (the order sequence of extraction and recovery, hash value, and feature bit matrix) can be stored effectively. Experimental results demonstrate the effectiveness of the proposed scheme.

**Acknowledgements** This research was funded by the Construction Project of Shanghai Key Laboratory of Molecular Imaging (Grant no. 18DZ2260400), the Fund from the Shanghai Municipal Education Commission, China (Class II Plateau Disciplinary Construction Program of Medical Technology of SUMHS, 2018–2020), and "Climbing" Program from SUMHS B3-0200-20-311007.

## References

1. Fridrich, J., Goljan, M., Du, R.: Lossless data embedding—new paradigm in digital watermarking. EURASIP J. Adv. Signal Process. **2002**(2), 185–196 (2002)
2. Tian, J.: Reversible data embedding using a difference expansion. IEEE Trans. Circuits Syst. Video Technol. **13**(8), 890–896 (2003)
3. Ni, Z., Shi, Y.-Q., Ansari, N., Su, W.: Reversible data hiding. IEEE Trans. Circuits Syst. Video Technol. **16**(3), 354–362 (2006)
4. Thodi, D.M., Rodriguez, J.J.: Expansion embedding techniques for reversible watermarking. IEEE Trans. Image Process. **16**(3), 721–730 (2007)
5. Zhang, X.: Reversible data hiding in encrypted images. IEEE Signal Process. Lett. **18**(4), 255–258 (2011)
6. Hong, W., Chen, T., Wu, H.: An improved reversible data hiding in encrypted images using side match. IEEE Signal Process. Lett. **19**(4), 199–202 (2012)
7. Qin, C., Zhang, X.: Effective reversible data hiding in encrypted image with privacy protection for image content. J. Vis. Commun. Image Represent. **31**, 154–164 (2015)
8. Hong, W., Chen, T.-S., Chen, J., Kao, Y.-H., Wu, H.-Y., Wu, M.-C.: Reversible data embedment for encrypted cartoon images using unbalanced bit flipping. Proc. Int. Conf. Swarm Intell. **7929**, 208–214 (2013)
9. Zhang, X.: Separable reversible data hiding in encrypted image. IEEE Trans. Inf. Forensics Secur. **7**(2), 826–832 (2012)
10. Wu, X., Sun, W.: High-capacity reversible data hiding in encrypted images by prediction error. Signal Process. **104**, 387–400 (2014)
11. Xiao, D., Xiang, Y., Zheng, H.Y., Wang, Y.: Separable reversible data hiding in encrypted image based on pixel value ordering and additive homomorphism. J. Vis. Commun. Image Represent. **45**, 1–10 (2017)
12. Qin, C., Zhang, W., Cao, F., Zhang, X., Chang, C.-C.: Separable reversible data hiding in encrypted images via adaptive embedding strategy with block selection. Signal Process. **153**, 109–122 (2018)
13. Fu, Y., Kong, P., Yao, H., Tang, Z., Qin, C.: Effective reversible data hiding in encrypted image with adaptive encoding strategy. Inf. Sci. **494**, 21–36 (2019)
14. Qin, C., Qian, X., Hong, W., Zhang, X.: An efficient coding scheme for reversible data hiding in encrypted image with redundancy transfer. Inf. Sci. **487**, 176–192 (2019)
15. Guan, B., Xu, D.: An efficient high-capacity reversible data hiding scheme for encrypted images. J. Vis. Commun. Image Represent. **66**, 102744 (2020)
16. Ge, H., Chen, Y., Qian, Z., Wang, J.: A high capacity multi-level approach for reversible data hiding in encrypted images. IEEE Trans. Circuits Syst. Video Technol. **29**(8), 2285–2295 (2019)
17. Puteaux, P., Puech, W.: An efficient MSB prediction-based method for high-capacity reversible data hiding in encrypted images. IEEE Trans. Inf. Forensics Secur. **13**(7), 1670–1681 (2018)
18. Wu, H.T., Yang, Z., Cheung, Y.M., Tang, S.: High-capacity reversible data hiding in encrypted images by bit plane partition and MSB prediction. IEEE Access **7**, 62361–62371 (2019)
19. National Electrical Manufacturers Association: Digital imaging and communications in medicine (DICOM). Part 5: Data Structures and Encoding (2020)
20. DICOM images (2020). https://pan.baidu.com/s/1eqebvvFihmCpkviwcVYb9w. Accessed 23 Nov 2020

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