A secure lossless recovery for medical images based on image encoding and data self-embedding

Hang Gao¹ · Tiegang Gao²

Received: 6 June 2021 / Revised: 19 September 2021 / Accepted: 6 October 2021 / Published online: 22 October 2021
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

To protect the security of data outsourced to the cloud, the tamper detection and recovery for outsourced image have aroused the concern of people. A secure tampering detection and lossless recovery for medical images (MI) using permutation ordered binary (POB) number system is proposed. In the proposed scheme, the region of interest (ROI) of MI is first extracted, and then, ROI is divided into some no-overlapping blocks, and image encoding is conducted on these blocks based on the better compression performance of JPEG-LS for medical image. After that, the generated compression data by all the blocks are divided into high 4-bit and low 4-bit planes, and shuffling and combination are used to generate two plane images. Owing to the substantial redundancies space in the compressed data, the data of each plane are spread to the size of the original image. Lastly, authentication data of two bits is obtained for every pixel and inserted into the pixel itself within the each plane, and the corresponding 10-bit data is transformed into the POB value of 8-bit. Furthermore, encryption is implemented on the above image to produce two shares which can be outsourced to the cloud server. The users can detect tampered part and recover original image when they down load the shares from the cloud. Extensive experiments on some ordinary medical image and COVID-19 image datasets show that the proposed approach can locate the tampered parts within the MI, and the original MI can be recovered without any loss even if one of the shares are totally destroyed, or two shares are tampered at the ration not more than 50%. Some comparisons and analysis are given to show the better performance of the scheme.

Keywords Image encoding · JPEG-LS · Permutation ordered binary · Tamper detection and recovery

1 Introduction

With the rapid development of network and communication, a large of number of image data are produced and outsourced to the cloud. As the cloud is regarded as dishonest entity, the privacy protection of outsourced image has drawn much attention from researchers, and many schemes have been proposed to solve the problem. In general, there are two approaches to protect privacy of the outsourced image; one is to conduct encryption of image before it is outsourced to the cloud [1–3], the other one is to

utilize the combination of encryption and digital watermarking to protect images [4–6].

As one kind of important image types, the protection of medical image has its particularity; it demands that when the medical image is recovered from its encrypted or watermarked form, the original image should be restored without any loss, as it has vital influence to medical diagnosis if there are some tamperers in the original image, based on these demands, people have pay much more attention to the location and lossless recovery of tamperers of original image, and many schemes have been presented [7–18]. For example, an improved tampering localization and self-recovery scheme based on fragile watermarking is presented by Sergio Bravo-Solorio, et al. [7], here, they use a secure block-wise mechanism to locate distorted blocks of pixels, and the tampering position accuracy is refined by means of an iterative pixel-wise mechanism, which estimates the original watermarked pixels in altered regions.

✉ Tiegang Gao
gaotiegang@nankai.edu.cn

¹ College of Computer Science, Nankai University, Tianjin 300381, China

² College of Software, Nankai University, Tianjin 300381, China

Afterwards, due to the good performance of compressed representation of Vector Quantization (VQ) and the automatic recovery capability of image inpainting, a new scheme is proposed to implement adaptive self-recovery for tampered digital images based on the two characteristics [8]. In order to provide detection and recovery rate for tamperers, a method based on singular value decomposition of the image blocks is proposed for digital image tamper localization and self-recovery, better performance is achieved [9]. Afterwards, a discrete cosine transform (DCT) based self-recoverable fragile watermarking scheme is presented, in this scheme, two authentication bits and ten recovery bits within the block are embedded in the three least significant bits (LSBs) of the block itself while recovery bits are embedded in the three LSBs of the corresponding mapped block [10]. In the meantime, they presented a self-embedding fragile watermarking scheme with effective image authentication and restoration based on a quantization and DCT [11]. In the scheme proposed by Sreenivas et al., the 4 bits authentication of 2×2 image block and restoration bits are embedded into the randomly selected distinct blocks by the help of chaos, and the better performance in tamperers detection and the recovery under extensive tampering is achieved [12]. In recent years, Gao et al. presented a reversible data hiding algorithm for medical images, the scheme achieved contrast enhancement of the region of interest (ROI), and in the meantime, it can locate the position of the tamperers against attacks on the ROI [13]. Tai et al. presented a tamperers detection and recovery scheme, in their approach, the fragile watermark of one block which consists of authentication data and recovery data are embedded into the other block according to the embedding sequence generated by chaotic map, and in the meantime, to reduce the smoothing blocking effect of recovered images, the wavelet transform are used to enhance the image contrast [14]. Not long after, a fragile and blind dual watermarking method for tamper detection and self-recovery based on the lifting wavelet and the halftoning technique is presented, and better performance was achieved [15]. Singh et al. proposed a self-embedding fragile watermarking technique based on quantization, and block truncation coding (BTC), and even if tempering rate is 50%, the reconstructed image has high peak signal-to-noise ratio (PSNR) [16]. Based on improved joint source-channel coding optimizer, Gu et al. gave a new image self-recovery scheme; the algorithm performs quadtree decomposition of original grayscale image corresponding to different decomposition factors and performs bit-plane layering according to the block class obtained by quadtree decomposition, and achieved high quality of restored image [17]. And recently, a authentication and self-recovery for color image is presented in [18], the novelty lie in that bilateral filter that

efficiently suppresses noise preserving image edges is used in the image post-processing.

Although better performances of image tamperers location and recovery have been obtained in the above schemes, one defect for most of them is that they can't recovery the original image without any loss with the tamperers happened. Therefore, some schemes using secret shares are proposed to solve the problem. Among them, an image sharing scheme that allows the user to retrieve a scaled or cropped version of the secret image by operating directly on the shadow images is presented in [19], and a secured image data sharing scheme over cloud domain based on the Shamir's secret sharing and permutation ordered binary number system is presented in [20]. After that, permutation ordered binary (POB) number system and Chinese remainder theorem (CRT) are used to generate multiple secret shares, and it has better performance for image reconstruction [21]. Recently, tamperers detection and recovery using POB aroused the attention of researchers; for example, Singh et al. proposed an image tampering detection and localization using POB number system, the scheme can accurately identify the altered pixels via authentication bits and localizes the tampered area, the tampered portion is also reflected back in the reconstructed image that is obtained [22]. Then, they presented an approach to detect and accurately recover tampered regions of video at the pixel level in the encrypted domain, the efficacy of the scheme has been validated by testing against various attack scenarios [23, 24]. Xiang et al. proposed a secure image tampering detection and self-recovery scheme based on POB and singular value decomposition (SVD), the scheme used authentication for tamperers detection and recovery watermarking for recovery of tamperers [25]. After that, Liu et al. gave a lossless image hierarchical recovery based on POB number system; the scheme contains a two-level comparison, neighbor-based refinement and watermark-based refinement in the stage of image recovery [26]. Recently, You et al. suggested a lossless self-recovery watermarking scheme with JPEG-LS compression scheme, the scheme implements JPEG-LS on ROI, and then embeds the compressed data in to the whole image, test results on medical images show the effectiveness of the proposed scheme.

It can be seen from the above secret shares based schemes that, most of them can't achieve totally reversibility for the recovered image with the shares tampered. To overcome this problem, a secure lossless tamperers detection and recovery approach for medical image (MI) is proposed; the proposed scheme utilizes the better compression performance of JPEG-LS and POB system, and achieves the totally reversible recovery of tamperers even two shares are tampered with size not more than 50%. Some comparisons and analysis are given to testify the

proposed scheme. The main contributions of this work are summarized as follows:

- Using JPEG-LS and the POB system, a lossless recovery and tamper detection approach for the original medical image is proposed.
- Utilizing the rich redundant features of data generated by the JPEG-LS, the proposed scheme achieves the strong robustness against tamperers of the shares. The original image can be restored without any loss even if one of the shares is totally tampered or both shares are tampered with the size less than 50%.
- Extensive experiments on medical images, including COVID-19 images show the better performance of tamperers detection for the proposed scheme.

The rest sections of this paper are organized as follows: In Sect. 2, some the preliminaries related to the proposed scheme are described. The detailed description of the proposed scheme and experimental results analysis are discussed in Sect. 3. Finally, some conclusions are drawn in Sect. 4.

2 Preliminaries

In this section, some preliminaries such as JPEG-LS-based image encoding and POB number system are first introduced, and some notations used in this paper are summarized in Table 1.

2.1 JPEG-LS-based image coding

JPEG-LS is a lossless or near lossless compression standard for continuous-tone image proposed by the International Standards Organization ISO/IEC JTC1 [28, 29], it is based on the LOCO-I algorithm (low complexity lossless compression for images) which depends on a pixel prediction based contextual statistical model [30]. It has been widely used for image compression and image security [31, 32].

2.2 POB number system

The POB number system is proposed by Sreekumar and Sundar [33]. The system is denoted by POB (n, r), where n and r are non-negative integral parameters such that $n \geq r$. In this number system, all the integers are in the range 0, 1, $\binom{n}{r} - 1$, a binary string with r 1's can be expressed by $B = b_{n-1}b_{n-2} \cdots b_0$. Then, the value of the POB-number B is calculated by following formula

$$V(B) = \sum_{j=0}^{n-1} b_j \binom{j}{p_j} \quad (1)$$

where, $p_j = \sum_{i=0}^j b_i$. It can be proved that, every such POB-number, $B = b_{n-1}b_{n-2} \cdots b_0$, there exists a distinct representation of POB value.

For example, POB value of the binary string with 10-bit 1,001,101,010 can be calculated by Eq. (1).

$$\begin{aligned} V(B) &= b_0 \times \binom{0}{0} + b_1 \times \binom{1}{1} + b_2 \times \binom{2}{1} \\ &\quad + b_3 \times \binom{3}{2} + b_4 \times \binom{4}{2} + b_5 \times \binom{5}{3} + b_6 \\ &\quad \times \binom{6}{4} + b_7 \times \binom{7}{4} + b_8 \times \binom{8}{4} + b_9 \times \binom{9}{5} \\ &= 0 \times 1 + 1 \times 1 + 0 \times 2 + 1 \times 3 + 0 \times 6 \\ &\quad + 1 \times 10 + 1 \times 15 + 0 \times 35 + 0 \times 70 + 1 \times 126 \\ &= 155 \end{aligned}$$

At present, several schemes have been proposed to show POB based method has better application in secret sharing and recovery of tampered regions in image [21–24].

2.3 Image shuffling

The aim of image shuffling is to strengthen the security of the proposed scheme. It can be expressed by formula (2)

$$(I_{shuf}, new_index) = \Phi(I, index, Key) \quad (2)$$

where, I is the original data which may consists of n

Table 1 Notations

Notation	Description	Notation	Description
MI	Medical images	BTC	Block truncation coding
POB	Permutation ordered binary	PSNR	Peak signal-to-noise ratio
ROI	Region of interest	CRT	Chinese remainder theorem
VQ	Vector quantization	SVD	Singular value decomposition
DCT	Discrete cosine transform	NPCR	Number of pixel changing rate
LSBs	Least significant bits	UACI	Unified averaged changed intensity

numbers to be shuffled, Key is the secret key used for generating new_index for shuffling, and the index is index sequences of pixel in the original I , while I_{shuf} is the shuffled I , and new_index is the new index sequences of numbers in original I produced by nonlinear mapping function Φ . In the proposed scheme, nonlinear mapping Φ is given by Logistic chaos mapping (LCM):

$$x_{n+1} = 4x_n(1 - x_n) \quad (3)$$

where, the initial value x_0 of LCM is used as the secret Key , which is not equal 0.5.

For example, the original image is assumed to be given by matrix A or transformed sequence B from A in column sequence.

$$A = \begin{bmatrix} 0 & 234 & 15 \\ 43 & 0 & 123 \\ 235 & 128 & 12 \end{bmatrix} B$$

$$= \{0, 43, 235, 234, 0, 128, 15, 123, 12\}$$

The initial value of LCM is given by $x_0^{(1)} = 0.003096800078798$, the produced sequence by iterating LCM is as followings:

$$x(1:9) = \begin{bmatrix} 0.012348839632280 & 0.048785383168064 & 0 \\ 0.604664580195915 & 0.956181302609652 & 0 \\ 0.558026272045897 & 0.986531807009822 & 0 \end{bmatrix}$$

Then, x is sorted by ascending order and the new sequence and new index are generated:

$$x_{new}(1:9) = \begin{bmatrix} 0.012348839632280 & 0.048785383168064 \\ 0.167594476597446 & 0.185621478228838 \\ 0.604664580195915 & 0.956181302609652 \end{bmatrix}$$

The new generated index sequence is $index_{new} = \{1, 2, 9, 6, 3, 7, 4, 5, 8\}$ the matrix A is rearranged according to the new generated index which is depicted in Fig. 1.

3 The proposed scheme

In this section, the detailed flowchart of the proposed scheme is first presented, and then, every procedure in the process will be described in detail. After that, some experimental results, analyses, comparisons and discussions are given to show the effectiveness of test for the

0, 43, 235, 234, 0, 128, 15, 123, 12	← Original sequence
1, 2, 3, 4, 5, 6, 7, 8, 9	← Original index
1, 2, 9, 6, 3, 7, 4, 5, 8	← New index generated by LCM
0, 43, 12, 128, 235, 15, 234, 0, 123	← Shuffled sequence

Fig. 1 Example of image shuffling

Fig. 2 The overall flowchart of the proposed scheme

proposed scheme. The overall flowchart of the proposed scheme is shown in Fig. 2.

3.1 Generation of shares

In order to generate the shares which are outsourced to the cloud, the original medical image need to be processed by several steps, and we will discuss these procedures in the followings.

3.1.1 Encoding image

In the proposed scheme, the ROI is first extracted automatically or by hand from the original medical image. After that, ROI is divided into some no-overlapping blocks with the size of 8×8 , and then JPEG-LS is used to encode every image block of ROI. All the compressed data of every block are combined into the compressed information

Fig. 3 The generation of high-level plane image and low-level plane image. **a** The generation of high-level plane image. **b** The generation of low-level plane image

which is denoted as I_{ced} . Then, the corresponding sequence number should be:

3.1.2 Bit-plane separation

In this section, bit-plane will be conducted for the compressed image data I_{ced} , the detailed descriptions of separation into two images are presented in the following steps:

- (1) Extract the high 4-bit for every number in the I_{ced} , and call it $high_p$, and in the meantime, the low 4-bit of the data is also extracted, and it is labeled as low_p .
- (2) Shuffle the $high_p$ with the key k_h and low_p with the key k_l , respectively. The shuffled data in high-level plane image is labeled as $high_{sp}$, and the shuffled data for low 4-bit is labeled as low_{sp} .
- (3) Generate high-level plane of 8-bit. In this processing, the 8-bit high-level plane image is produced by $high_p = 16 \times low_{sp} + high_p$.
- (4) Generate low-level plane image of 8-bit by $low_p = 16 \times high_{sp} + low_p$.

Fig. 4 The example for authentication embedding and transformation

Thus, the two bit plane images are produced through the above steps. The flowchart for the generation of high-level plane and low-level plane can be shown in Fig. 3.

3.1.3 The generation and self-embedding of authentication data

In this stage, for the above generated bit-plane images $high_p$ and low_p , for every data, one 2-bit authentication data will be produced and embedded into the bit plane image data itself. The procedure is described by the following methods.

For the given data a in $high_p$, assume it is expressed by $b_7b_6b_5b_4b_3b_2b_1b_0$, where $b_j, j = 0, 1, \dots, 7$ is 0 or 1. If the number of 1 within $b_7b_6b_5b_4$ is odd, the first bit is set to be 1; else it is set to be 0. In the same way, the second bit is set to be 1 if and only if the number of 1 within $b_3b_2b_1b_0$ is odd.

In this way, 2-bit data is generated by a , and then it is appended at the end of data a , thus, the data turns to become 10-bit data. In order to easily and conveniently process the image, every data with 10-bit will be transformed into the corresponding decimal value of POB number with the size of 8-bit using formula (1). An example is illustrated in Fig. 4.

In the same way, the authentication data in low_p can be obtained and embedded in the data itself, and finally it is transformed into the POB value with size of 8-bit. The detailed description on the generation and self-embedding of authentication can be shown in Fig. 5.

Fig. 5 The generation of authentication data and embedding

Fig. 6 The sketch map of spreading of compressed data

Fig. 7 Some typical test images used for experiments

After this procedure, two images with embedded authentication data are generated, they are labelled as $high_e$ and low_e , respectively.

3.1.4 Spreading of two level data and encryption

Here, for the above generated bit plane images which are self-embedded by authentication data, we will spread image data into the same size as that of the original image. As the compressed data is shorter than the original image, some data in $high_e$ or low_e may appear several times in the spreading image.

Figure 6 give an illustration on the spreading of compressed data. In order to produce the same length of data as that of the original image, the compressed data need to be put several times to achieve the same size as that of the original image.

For example, the original medical image IM-0001-0012 shown in Fig. 7a is with the size of 512×512 , there are total 262,144 pixels. For the ROI shown in Fig. 7b, after compression, the number of data in high level plane is 65833, thus, every data in generated bit plane image $high_e$ may emerge $262144/65833=3.98$ times when they are

spread into the size of 512×512 . In the same way, there are 84,598 data in the generated bit plane image generated from Fig. 7d which is the ROI of Fig. 7c, thus, when the data are spread to the size of 512×512 , every data may appear $\text{floor}(262144/84598) = 3.09$ times.

Lastly, for the safety, the two bit plane images after spreading are encrypted by hyper-chaos based encryption algorithm [34]. It should be mentioned that, in this stage, we can use any kind of secure encryption algorithm. Thus, two shares are produced; they can be outsourced to the same cloud server or different cloud server.

3.2 Tamper detection and cover image recovery

When the users of the medical image download the shares from the cloud server, they can use the following steps to verify the integrity of the medical image and recover the original secret image.

- (1) For the two shares $high_{es}$ and low_{es} , they are firstly decrypt by the same secret key as that used in encryption process, and two shares $high_s$ and low_s are got.

- (2) Convert $high_s$ and low_s into the POB numbers with 10-bit by Algorithm 2 shown in Fig. 8. The returned image is 10-bit data.
- (3) For the image $high_s$, the last two bits data from every 10-bit data are extracted. Thus, one can calculate the number of 1 with the high 4-bit and that of the low 4-bit except for the two least significant bits for every data, thus two groups data of 2-bit data are obtained, respectively. One can judge whether the pixel value is tampered or not through comparing the obtained two group's data. The same operation can also be performed on low_s .

After removing the last two authentication bits, high bit plane image $high_p$ and low bit plane image low_p are derived.

- (4) Recovery of the original image

For the obtained high bit plane image $high_p$, we separate it into high 4-bit and low 4-bit plane, called $high_{hbitp}$ and $high_{lbitp}$, and then reshuffle the $high_{hbitp}$ to get $high_{shbitp}$ using the same key k_h as that used in original shuffling. After that, the compressed data can be restored by formula (4).

$$Compresseddata_h = 16 \times high_{lbitp} + high_{shbitp} \quad (4)$$

Lastly, the compressed data $Compresseddata_h$ are decoded using JPEG-LS block by block to restore the original ROI, and the original image can be losslessly recovered.

In the same way, the original image can also be recovered through low_p . Low bit plane image low_p is separated into high 4-bit and low 4-bit plane, called low_{hbitp} and low_{lbitp} , and then reshuffle the low_{hbitp} to get low_{shbitp} using the same key k_l as that used in original shuffling. After that, the compressed data can be restored by formula (5).

$$Compresseddata_l = 16 \times low_{shbitp} + low_{lbitp} \quad (5)$$

The generated $Compresseddata_l$ then are decoded by JPEG-LS block by block to restore the original ROI, and thus the original image can be losslessly recovered.

(5) Tamper detection

For the two shares which are outsourced to the cloud, once one or two shares are tampered, users who want to use the image must be able to know whether the two shares have been tampered or

Fig. 8 Description of the algorithm 2

Algorithm 2: [POBN: Generate POB number corresponding to the given POB value]

Input: n , r and v , where $r \leq n$ and $0 \leq v \leq \binom{n}{r} - 1$.

Output: The POB number $B = b_{n-1}b_{n-2} \dots b_1b_0$.

Step 1: Let $j = n$ and $temp = v$.

Step 2: **for** $k = r$ down to 1 **do**:

1. **repeat** {
2. $j = j - 1$;
3. $p = \binom{j}{k}$
4. **if** ($temp \geq p$)
5. $temp = temp - p$;
6. $b_j = 1$;
7. **else** $b_j = 0$;
8. **}** **until** ($b_j = 1$);

Step 3: **if** ($j > 0$)

for $k = j - 1$ down to 0 **do**:

$b_k = 0$;

Step 4: **return** B ;

Algorithm 3: [Refine-Data: Refine data from the 10-bits POB numbers of the high bit plane image]

Input: $High_p$

Output: Extracted compressed data CompressedDH and detection results Detect_H.

Define: (1) M, N is the length and width of original image. L is the length of the compressed data

(2) CompressedDH and Detect_H are returned compressed data and detection results with size of L

```

1.  Read  $High_p, Low_p$ .
2.  Number  $\leftarrow \text{floor}(\frac{N \times M}{L})$  % Number stand for the times every data appears
3.  for j = 1 up to L do:
4.    repeat {
5.       $Hb \leftarrow High_p^{2bit}(j)$  % the last 2-bit data from the  $j^{th}$  data in  $High_p$ 
6.       $CHb \leftarrow \text{Compute}(High_p^{8bit}(j))$  % calculate the 2-bit from the high 8-bit
      of  $High_p$ 
7.      If ( $CHb < Hb$ ) % it means the pixel is tampered
8.         $Hb \leftarrow High_p^{2bit}(j + L)$ 
9.         $CHb \leftarrow \text{Compute}(High_p^{8bit}(j + L))$ 
10.       If ( $CHb < Hb$ )
11.         ...
12.         If ( $CHb < Hb$ )
13.            $Hb \leftarrow High_p^{2bit}(j + (Number - 1) \times L)$ 
14.            $CHb \leftarrow \text{Compute}(High_p^{8bit}(j + (Number - 1) \times L))$ 
15.           If ( $CHb < Hb$ )
16.             Detect_H(j)=1
17.             CompressedDH(j)=  $High_p^{8bit}(j)$  % Extracted data
18.           Else
19.             Detect_H(j)=0
20.             CompressedDH(j)=  $High_p^{8bit}(j)$  % Extracted data
21.           End
22.         Else
23.           Detect_H(j)=0
24.           CompressedDH(j)=  $High_p^{8bit}(j)$  % Extracted data
25.         End
26.       Else
27.         Detect_H(j)=0
28.         CompressedDH(j)=  $High_p^{8bit}(j)$  % Extracted data
29.       End
30.     Else
31.       Detect_H(j)=0
32.       CompressedDH(j)=  $High_p^{8bit}(j)$  % Extracted data
33.     End
34.   } until (j > L);
35. end

```

Step 4: return CompressedDH, Detect_H;

Fig. 9 Description of the algorithm 3

Fig. 10 The flowchart of recovery for the compressed data (a). Recovery of compressed data from high bit plane image (b). Recovery of compressed data from low bit plane image

original image can be restored or not. Thus, the following sections will describe how the tampered blocks are detected and how the original image is recovered.

As the compressed data are spread over the entire share, some data may be the same in different locations. Thus, in order to get the real compressed data, these data in different positions need to be judged and compared, the refinement processes of extracting and detection compressed data from $high_p$ are described in the followings.

- (1) For the obtained high bit plane image $high_p$ and low bit plane image low_p with the 10 bits data, The number k of occurrences of each data is firstly calculated, thus there are k groups identical compressed data in $high_p$.
- (2) Assume that the length of compressed data is L , for the data in one group of $high_p$, calculate the 2 bits authentication data and extract last 2-bit data, if these two 2 bits data are not equal, repeat conducting the same operation for the same data in the next group; If the extracted 2 bits data and 2 bits authentication data within

all the groups are not equal, the data detection result is set to 1 in this position; else the detection result is set to be 0, and the extracted data in $high_p$ is saved. In this way, the data without being tampered for compressed data are extracted, only data in all the groups are tampered, it is regarded as a tampered one. The detailed descriptions of detection and extraction are given by Algorithm 3 shown in Fig. 9. It can be seen that the core of the Algorithm 3 is to obtain the real compressed data, as there are several copies for every compressed data; we only select the one which is not tampered as the real one. For every compressed data, its detection result is kept in $Detect_H$, where, the value 0 means the compressed data is real; else it means the compressed data is tampered.

The compressed data and detection results from the Low_p with 10-bit can also be obtained by the same way as that used in Algorithm 3; they are denoted by $CompressedDL$ and $Detect_L$, respectively.

Fig. 11 The images used for test **Fig. 12** The image and comparison of histograms with other method

- (6) Data refinement between the compressed data produced by two bit plane images

After two groups compressed data are derived from Low_p and $High_p$, the data refinement between these two groups of data must be conducted to obtain real and effective compressed data.

Firstly, for the share CompressedDH, it is first separated into two 4-bit plane data, and then the high 4-bit plane and the Detect_H are reshuffled using the same secret key as that used in the process of generation of shares. Then for every data, we compare the value of Detect_H and shuffled Detect_H, if they are equal to zero, then this data is considered to be intact, else it is regarded as tampered.

In the same way, for the CompressedDL and $Detect_L$, the same method can be used to derive real and effective data from the Low_p . The process flowcharts are shown in Fig. 10.

Lastly, we should compare detection results of $Detect_L$ and $Detect_H$, the compressed data is regarded as intact if and only if some one of the value in $Detect_L$ and $Detect_H$ is 0, and the recovered compressed data is replaced with the corresponding detection result.

Lastly, the original ROI can be restored by JPEG-LS decoding block by block without any loss.

4 Experimental results and analysis

In this section, a large numbers of experiments for the proposed scheme are implemented, the images used for test are download from some open dataset with the size of 512×512 . Some typical images are shown in Fig. 11. In the meantime, some comparisons on embedding rate (bpp), visual quality of the final recovered image are given to testify the proposed scheme.

4.1 Security analysis

- (1) Key space analysis

It can be seen from the description of the proposed scheme, the generation of anyone share is relevant to the shuffling algorithm, and the key space for shuffling is 10^{14} . In the meantime, from the definition of POB number system, a specific POB number with 10-bit can be represented in the range of $\left[0, \binom{n}{r} - 1\right]$, here, r is the number of 1s, and n is 10. Obviously, if r is set to be 4, POB number has 210 possibilities, thus, for image with size of $N \times M$, the successful probability of guessing the share is $\left(\frac{1}{210}\right)^{M \times N}$. Lastly, the key space used for encryption is 2^{128} , so the proposed scheme can effectively resist brute attack.

On the other hand, different from the scheme [22, 25] which generate shares by POB number system in the last step of the scheme, in the last step of our scheme, we implement encryption of the shares generated by POB, this is because that

Fig. 13 The experimental results. **a** Original image. **b** Extracted ROI. **c** and **d** shares. **e** Restored image

shares generated by POB has some kind of statistics features that may arouse the curiosity of attackers.

For example, histogram of image can embody the statistics feature of pixel distribution, Fig. 12a is an original medical image, the histogram of one share generated by our scheme is shown in Fig. 12b, however, Fig. 12c is the histogram generated by reference [26]. Obviously, there is no any visual statistics feature to be observed in Fig. 12b, however, the feature of Fig. 12c can cause suspicions of others.

(2) Image entropy

Information entropy can be used to find the unpredictability and randomness, and the ideal value for random image for which the pixel values always fall in the range of 0–255 is always equal to 8. The entropy of image is given by formula (8) according to Shannon's communication theory of secrecy systems.

$$H = - \sum p_i \log p_i \quad (6)$$

where p_i is the probability of the pixel values in the image and summation is taken over all possible pixel values. It has been shown that the entropy of a pure random noise image should be equal to 8, and the entropy is 7.7706 in the case of 2-bit authentication [25].

In the proposed scheme, the entropy of share is 7.9992; hence, it is almost equivalent to a random image. If we first encrypt the bit plane image, and then embed the authentication data, the entropy is 7.21872.

(3) The correlation coefficients

The correlation coefficients between adjacent pixels reflect the image feature, the smaller correlation coefficients of encrypted image mean that the encrypted image looks like noise image, it can be calculated as follows:

$$r = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\left(\sum_{i=1}^N (x_i - \bar{x})\right)^2 \left(\sum_{i=1}^N (y_i - \bar{y})\right)^2}} \quad (7)$$

where, x_i and y_i here are the values of the adjacent pixels, $\bar{x}$ and $\bar{y}$ are the means of them, N is the number of the adjacent pixels.

For the proposed scheme, the correlation coefficients in vertical, horizontal and diagonal direction are 0.01137, 0.0162, and 0.0044, respectively.

(4) The number of changing pixel rate (NPCR) and the unified averaged changed intensity (UACI)

NPCR and UACI are two common quantities used to evaluate the strength of image encryption algorithms with respect to differential attacks. They signify the chance of occurrence of the attack and its sensitivity towards the source image by changing the value. They are given by Eqs. (8) and (9), respectively.

$$NPCR = \frac{\sum_{i=1}^N d_i}{N} \times 100 \quad (8)$$

$$d_i = \begin{cases} 1 & I(i) \neq I'(i) \\ 0 & I(i) = I'(i) \end{cases} \quad i = 1, 2, \dots, N$$

$$UACI = \frac{1}{N} \sum_{i=1}^N \frac{|I(i) - I'(i)|}{255} \quad (9)$$

For the proposed scheme, the $NPCR$ is 99.6033 and the $UACI$ is 33.4599, this means that the proposed method can well resist the known-plaintext and chosen-plaintext attacks.

It is obvious; the above test results show that the proposed scheme can effectively resist brute force attack as it has large space of secret key. And moreover, it can also resist attack coming from statistical feature of the shares.

Fig. 14 The test results against some attacks on the share.

a Text addition attack.
b Content removal attack.
c Content replacement attack;
d–f the detection results of the first round; **g–i** are the the detection results of the second round; **j–l** are the losslessly recoverd image

Table 2 Result of lossless recovery for original image

Tampered area in high-bit plane	Tampered area in low-bit plane	Whether the original image is losslessly recovered
10%	Any size	✓
30%	Any size	✓
50%	Any size	✓
80%	No tamperers	✓
100%	No tamperers	✓

4.2 Tampering localization and image recovery

Firstly, the medical image IM-0001-0012 is used for various tests. In the experiments, if there is no any tampering in the two shares, the original ROI can be totally recovered without any loss. The original image, ROI, generated two shares and the recovered ROI are shown in Fig. 13.

Next, some attack tests on shares are conducted. The attacks by text addition, content removal and content exchange on one share are shown in Fig. 14a–c. For the content removal attack, a block with size of 287×283 from the left corner point (95, 125) is cropped within the share. For the content exchange, a block with size of 312×321 from the left corner 104×110 is replaced by other contents. In this case, only one share is tampered, and the detection results of the first round are shown in Fig. 14d–f, and that of the second round are shown in Fig. 14g–i, the lastly refined detection results are shown in Fig. 14j–l. It has been shown that the finally restored ROI is the same as the original one. The original image can be losslessly recovered.

Furthermore, from the depiction of the proposed scheme, even if anyone of the two shares is totally tampered, the original image can be recovered without any loss; in the meantime, when one share is tampered with the size of less than 50%, whether what is the tampered size in the other share, the original image can be recovered without any loss. The experiment results are summarized in Table 2.

Next, some tests are given to verify the recovery performance of the proposed scheme with the two shares tampered. In the experiments, some contents removal attacks are given, and the test results are shown in Fig. 15. Where, Fig. 15a and b are tampered shares, and one-fourth of the share is tampered for every share, the recovered images by the algorithm [26] are given by Fig. 15c and d shows the totally recovered images by the proposed scheme. It can be seen that the proposed scheme has great improvement than reference [26] with the two shares tampered.

Lastly, some tests with no regular tamperers in the shares are presented. The experimental results are displayed in Fig. 16. Where, Fig. 16a is the original ROI, Fig. 16b and c are tampered shares with fifty percent of the size in one share, and the recovered ROI without any loss is shown in Fig. 16d. For the second row, Fig. 16e is the original ROI, Fig. 16f and g are tampered shares with different size, their tampered parts are greater than 35%, and the totally recovered ROI is shown in Fig. 16h.

4.3 Performance evaluations and comparisons

In order to evaluate the performance of the proposed scheme, some widely used evaluation criteria are first introduced.

The first criterion is the Peak Signal to Noise Ratio (PSNR) which is used to evaluate the degree of distortion of the image. It is defined by formula (10)

$$PSNR = 10 \log_{10} \frac{255^2}{MSE} \quad (10)$$

$$MSE = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N (I(i,j) - R(i,j))^2$$

where, M and N are length and width of the image, respectively. $I(i,j)$ and $R(i,j)$ are the pixel values of original image and restored image, respectively.

It can be seen from the above experimental results, when one share is tampered, no matter what is the ratio of the tampered parts, the original image can be totally recovered from the two shares. Here, we can observe the experimental results and some data from the relative schemes in reference [22–26] when different rations content removals are conducted. The experimental results are summarized in Table 3.

However, when the two shares are tampered, we have test more than fifty medical images, and find that as long as the two shares are tampered with the size of not more than 50% at any position, the original ROI can be recovered without any loss. Some comparisons with some typical algorithms are given in Table 4.

Fig. 15 The test results with the two shares tampered. **a** Attack on the first share; **b** Attack on the second share; **c** Recovery results by reference [26]; **d** The losslessly recovered image by the proposed scheme

Fig. 16 Test results with no regular tampers on the two shares

Table 3 The comparisons of PSNR with different ration content removal on one share

PSNR	Scheme	10%	20%	30%	50%
Recovered image	POB-based secret sharing [22]	$< \infty$	$< \infty$	$< \infty$	$< \infty$
	Video tamper recovering scheme [23]	$< \infty$	$< \infty$	$< \infty$	$< \infty$
	Video tamper detection based on POB [24]	$< \infty$	$< \infty$	$< \infty$	$< \infty$
	Image self-recovering based on POB [25]	43.37	40.12	38.24	36.17
	Image hierarchical recovery using POB [26]	∞	∞	∞	∞
	Image self-recovering with JPEG-LS compression [27]	$< \infty$	$< \infty$	$< \infty$	$< \infty$
	The proposed scheme	∞	∞	∞	∞

Table 4 The comparisons of PSNR with two share tampered of different ratios

PSNR	Scheme	10%	20%	30%	50%
Recovered image	POB-based secret sharing [22]	$< \infty$	$< \infty$	$< \infty$	$< \infty$
	Video tamper recovering scheme [23]	$< \infty$	$< \infty$	$< \infty$	$< \infty$
	Video tamper detection based on POB [24]	$< \infty$	$< \infty$	$< \infty$	$< \infty$
	Image self-recovering based on POB [25]	$< \infty$	$< \infty$	$< \infty$	$< \infty$
	Image hierarchical recovery using POB [26]	$< \infty$	41.23	36.31	30.23
	The proposed scheme	∞	∞	∞	∞

Fig. 17 Some medical images with more areas of ROI

Table 5 Comparison of the proposed method with other ones

Criteria	[22]	[23]	[24]	[25]	[26]	Proposed
Mode of recovery	None	POB numbers of compacted pixel values	None	Reconstruction + refinement	Two-level comparison + two refinements	Two-level comparison
Recovered image quality	Lossless	Lossless	None	Lossless	Lossless	Lossless
Blind authentication	Yes	Yes	Yes	Yes	Yes	Yes
Level of tamper detection	Pixel of a share	Pixel	Pixel	Block	Block	Pixel
Level of recovery	Block	Block	Block	Block	Block	Block

4.4 Analysis and comparisons of the proposed scheme with other approaches

In this section, we firstly discuss the strength and weakness of the proposed scheme, and then some comparisons with other approaches are given.

4.4.1 Discussions and analysis of the proposed scheme

In the proposed scheme, even if one share is totally tampered, the original ROI of medical image can also be recovered with the another share tampered by not more than 50%, this has been testified by experiments.

Next, the theoretical analysis will be given to support the experimental results. Here, we test more than 500 medical images from open datasets, including 100 medical images of bone [35] and 400 COVID-19 images [36–38]. The average number of produced data for our test after compression is half of the total number of pixels in original image, even if ROI is the image itself. For example, some images with more areas of ROI are shown in Fig. 17, among them, although there are large areas ROI in image, the ratios between the number of generated data after compression and the numbers of original image pixels are (a) 50.1, (b) 49.5, (c) 36 and (d) 32%, respectively. The average value is 41.9%. Moreover, some typical COVID-

19 images with large areas ROI are shown in Fig. 17e–h, and the same results are obtained. Hence, it is reasonable that even half of one share is tampered and another share totally is tampered, the original ROI can be restored without any loss.

Obviously, the strength of the proposed scheme lies in that a large amount redundant space is obtained after compression of the original image, these spaces can be used to put the compressed data several times, this kind of operation resembles embedding recovery watermarking in the compressed data. When original compressed data need to be recovered, the same data in multiple positions can be used for comparisons and the correct one will be kept according to voting rules. This can reduce the computation complexity of the algorithm.

On the other hand, although share tampering is detected in pixel level, as JPEG-LS can recover original image from the compressed data in sequence, this means that if one value in compressed data is detected to be tampered, the original image pixels of entire block will not be recovered. This is the weakness of the proposed scheme; this also means that the performance becomes better with the size of the block is smaller.

4.4.2 Some comparisons of the proposed scheme with other approaches

Among the entire recently proposed image tampering detection localization and recovery schemes based on secret sharing [22–26], as only the authentication bits are used by Singh in [22], this results in that self-recovery of tampering is not achieved. However, as authentication and recovery bits are embedded into the shares, thus, tampering detection and lossless self-recovery can be obtained in [23–26].

In the same way, the proposed approach doesn't use the recovery watermarking; it is based on JPEG-LS compression, and utilizes the characteristic of JPEG-LS to full advantage that large amounts of redundant space are generated through compression. Thus, these redundant spaces can be used for multi-data embedding, so as to be able to recover the tampered data. Other criteria used to compare with some existing algorithms are as follows:

(1) Mode of recovery

The mode describes how the original image is recovered.

(2) The quality of the recovered image

It can be classified into three types. One is the lossless recovery, it means the recovered image is the same as the original one; another one is called 'slightly lossy' recovery, it means that the PSNR value of the recovered image is near 40, other types are called lossy recovery.

(3) Blind authentication

It is called blind authentication if only the shares information is used for detection of the tampering.

(4) Level of tampering detection and recovery

It refers to point of view of the tampering detection, if the detection takes image as a unit, it is labelled as 'image'; if the unit of detection is block, it is labelled by 'block', and it is given 'pixel' if it takes pixel as a detection unit. It is the same for the recovery of image.

It can be seen from Tables 2, 3, 4 and 5, the proposed scheme uses the simple algorithm to achieve the better performance in tampering detection and recovery. Compared with the newly proposed approach [26], our method has some improvements in the three aspects.

(1) The proposed scheme can achieve lossless recovery even if the size of tampered parts is greater than 25% and less than 50%, this can't be obtained in reference [26]. This is mainly due to the usage of JPEG-LS encoding.

(2) Owing to that only authentication bit is used in the proposed scheme, two-level refinements are enough to recover the original image. However, two-level comparisons and two refinements need to be operated to recover the original image in reference [26].

(3) The proposed scheme can recover the original image without any loss with one share tampering not more than 50%, and the other share with any size of tampering. But this can't be achieved in [26]. This is because we utilize the ROI features of medical images and better performance of JPEG-LS for medical images.

5 Conclusion

This paper proposes a new detection and recovery of tampered parts in medical image using POB and JPEG-LS. The approach achieves the better performance of detection and recovery even with the one share is totally tampered or two shares are tampered with size of no more than 50% in the share. A large numbers of experiments are given to show that the scheme proposed here can be used for recover the original MI outsourced into the cloud without any loss. It is obvious that performance of the proposed scheme is directly bound up with the ability of JPEG-LS compression for medical image and POB. Thus, the future work will be focused on the performance improvement of the JPEG-LS algorithm, such that the original image can be recovered even when the compressed data is slightly modified. In the meantime, some efforts will be devoted to implementation of parallel processing algorithm for the

proposed scheme such that it can be suitable for large-scale medical image protection.

Acknowledgements This work was supported by the National Science and Technology Major Project, China (Grant No. 2018YFB0204304).

Author contributions HG conducted the experiments, performed the data analyses and wrote the manuscript; TG performed the analysis with constructive discussions.

Data availability All data, models, and codes generated or used during the study are available from the corresponding author by request. (gaotiegang@nankai.edu.cn).

Declarations

Conflict of Interest 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

- Gao, H., Gao, T.: Double verifiable image encryption based on chaos and reversible watermarking algorithm. *Multimed. Tools Appl.* **78**, 7267–7288 (2019)
- Tahir, M., Sardaraz, M., Mehmood, Z., et al.: CryptoGA: a cryptosystem based on genetic algorithm for cloud data security. *Clust. Comput.* **24**, 739–752 (2021)
- Jiang, L., Cao, Y., Yuan, C., Sun, X., Zhu, X.: An effective comparison protocol over encrypted data in cloud computing. *J. Inf. Secur. Appl.* **48**, 102367 (2019)
- Yao, Y., Zhang, W., Wang, H., Zhou, H., Yu, N.: Content-adaptive reversible visible watermarking in encrypted images. *Signal Process.* **164**, 386–401 (2019)
- Haddad, S., Coatrieux, G., Cozic, M., Bouslimi, D.: Joint watermarking and lossless JPEG-LS compression for medical image security. *IRBM* **38**(4), 198–206 (2019)
- Kahlessenane, F., Khaldi, A., Kafi, R., et al.: A robust blind medical image watermarking approach for telemedicine applications. *Clust. Comput.* **24**, 2069–2082 (2021)
- Bravo-Solorio, S., Nandi, A.K.: Secure fragile watermarking method for image authentication with improved tampering localization and self-recovery capabilities. *Signal Process.* **91**, 728–739 (2011)
- Qin, C., Chang, C.-C., Chen, K.-N.: Adaptive self-recovery for tampered images based on VQ indexing and inpainting. *Signal Process.* **93**, 933–946 (2013)
- Ansari, I.A., Pant, M., Ahn, C.W.: SVD based fragile watermarking scheme for tamper localization and self-recovery. *Int. J. Mach. Learn. Cyber.* **7**, 1225–1239 (2016)
- Singh, D., Singh, S.K.: Effective self-embedding watermarking scheme for image tampered detection and localization with recovery capability. *J. Vis. Commun. Image Represent.* **38**, 775–789 (2016)
- Singh, D., Singh, S.K.: DCT based efficient fragile watermarking scheme for image authentication and restoration. *Multimed. Tools Appl.* **76**, 953–977 (2017)
- Sreenivas, K., Kamakshiprasad, V.: Improved image tamper localization using chaotic maps and self-recovery. *J. Vis. Commun. Image Represent.* **49**, 164–176 (2017)
- Gao, G., Wan, X., Yao, S., Cui, Z., Zhou, C., Sun, X.: Reversible data hiding with contrast enhancement and tamper localization for medical images. *Inf. Sci.* **385–386**, 250–265 (2017)
- Wei-Liang, T., Zi-Jun, L.: Image self-recovery with watermark self-embedding. *Signal Process.* **65**, 11–25 (2018)
- Haghighi, B.B., Taherinia, A.H., Harati, A.: TRLH: Fragile and blind dual watermarking for image tamper detection and self-recovery based on lifting wavelet transform and halftoning technique. *J. Vis. Commun. Image Represent.* **50**, 49–64 (2018)
- SinghSingh, D.S.K.: Block truncation coding based effective watermarking scheme for image authentication with recovery capability. *Multimed. Tools Appl.* **78**, 4197–4215 (2019)
- Gu, Y., Yang, H., Yan, B., Wang, X., Zhao, Z.: Digital image self-recovery algorithm based on improved joint source-channel coding optimizer. *Multimed. Tools Appl.* **78**, 21041–21064 (2019)
- Molina-Garcia, J., Garcia-Salgado, B.P., Ponomaryov, V., et al.: An effective fragile watermarking scheme for color image tampering detection and self-recovery. *Signal Process.* **81**, 115725 (2020)
- M. Mohanty, W.T. Ooi, P.K. Atrey, Scale me, crop me, know me: supporting scaling and cropping in secret image sharing. In: Proceedings of the IEEE International Conference on Multimedia and Expo, 2013. doi: <https://doi.org/10.1109/ICME.2013.6607567>.
- P. Singh, N. Agarwal, B. Raman, ACM, Don't see me, just filter me: towards secure cloud based filtering using Shamir's secret sharing and POB number system. In: Proceedings of the tenth Indian conference on computer vision, graphics and image processing. 2016. doi: <https://doi.org/10.1145/3009977.3010036>.
- Deepika, M.P., Sreekumar, A.: A novel secret sharing scheme using POB number system and CRT. *Int. J. Appl. Eng. Res.* **11**(3), 2049–2054 (2016)
- Singh, P., Raman, B., Agarwal, N., Atrey, P.K.: Secure cloud-based image tampering detection and localization using POB number system. *ACM Trans. Multimed. Comput. Commun. Appl.* **13**(3), 1–23 (2017)
- Singh, P., Atrey, P.K.: Recovering tampered regions in encrypted video using POB number system. *Signal Process.* **74**, 96–109 (2019)
- Singh, P., Raman, B., Agarwal, N.: Toward encrypted video tampering detection and localization based on POB number system over cloud. *IEEE Trans. Circuits Syst. Video Technol.* **28**(9), 2116–2130 (2018)
- Xiang, Y., Xiao, D., Wang, H., Li, X.: A secure image tampering detection and self-recovery scheme using POB number system over cloud. *Signal Process.* **162**, 282–295 (2019)
- Liu, Y., You, Z., Gao, T.: Lossless image hierarchical recovery based on POB number system. *Signal Process.* **167**, 107293 (2020)
- You, Z., Liu, Y., Gao, T.: A lossless self-recovery watermarking scheme with JPEG-LS compression. *J. Inf. Secur. Appl.* **58**, 102733 (2021)
- JPEG-LS, Lossless. "Near-Lossless Coding of Continuous Tone Still Images." ISO/IEC JTC1/SC 29.
- Dimpal Bhimani, Pratap N. Chaurasiya, Bhavin Sedani, Optimized lossless image compression algorithm LOCO-I for small images. In: Proceedings of 2016 Conference on Advances in Signal Processing (CASP), 2016.
- Afjal, M.I.: Md Al Mamun, Md Palash Uddin, Band reordering heuristics for lossless satellite image compression with 3D-CALIC and CCSDS. *J. Vis. Commun. Image Represent.* **59**, 514–526 (2019)
- Caldelli, R., Filippini, F., Barni, M.: Joint near-lossless compression and watermarking of still images for authentication and tamper localization. *Signal Process.* **21**(10), 890–903 (2006)

32. Haddad, S., Coatrieux, G., Cozic, M., Bouslimi, D.: Joint watermarking and lossless JPEG-LS compression for medical image security. *IRBM* **38**(4), 198–206 (2017)
33. Singh P, Agarwal N, Raman B. Don't see me, just filter me: towards secure cloud based filtering using Shamir's secret sharing and POB number system. In: Proceedings of the Tenth Indian Conference on Computer Vision, Graphics and Image Processing. ACM, 2016: 12.
34. Zhang, S., Gao, T.: A coding and substitution frame based on hyper-chaotic systems for secure communication. *Nonlinear Dyn.* **84**, 833–849 (2016)
35. <https://isbweb.org/data/vsj/humeral/>
36. COVID-9 Database: <https://www.sirm.org/category/senza-categoria/covid-19/>
37. <https://radiopaedia.org/> Accessed April 9, 2020.
38. COVID-19 Radiography Database: <https://www.kaggle.com/taw-sifurrahman/covid19-radiography-database/>

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

Tiegang Gao is a professor of college of software since 2006 in Nankai University, China. His research field includes information security, software engineering. He has authored or co-authored more than 100 papers in the relative fields.

Hang Gao is a PhD candidate, his research field is digital image processing and software engineering. He has published more than 10 papers in the field.