A reversible data hiding scheme in JPEG bitstreams using DCT coefficients truncation

Mingming Zhang[1*], Quan Zhou[1] and Yanlang Hu[1]
[1]National Key Laboratory of Science and Technology on Space Microwave, Xi'an Institute of Space Radio Technology
Xi'an, Shaanxi 710100 - China
[e-mail: zmm504@163.com]
*Corresponding author: Mingming Zhang

Received April 13, 2019; revised August 7, 2019; accepted September 9, 2019; published January 31, 2020

Abstract

A reversible data hiding scheme in JPEG compressed bitstreams is proposed, which could avoid decoding failure and file expansion by means of removing of bitstreams corresponding to high frequency coefficients and embedding of secret data in file header as comment part. We decode original JPEG images to quantified 8×8 DCT blocks, and search for a high frequency as an optimal termination point, beyond which the coefficients are set to zero. These blocks are separated into two parts so that termination point in the latter part is slightly smaller to make the whole blocks available in substitution. Then spare space is reserved to insert secret data after comment marker so that data extraction is independent of recovery in receiver. Marked images can be displayed normally such that it is difficult to distinguish deviation by human eyes. Termination point is adaptive for variation in secret size. A secret size below 500 bits produces a negligible distortion and a PSNR of approximately 50 dB, while PSNR is also mostly larger than 30 dB for a secret size up to 25000 bits. The experimental results show that the proposed technique exhibits significant advantages in computational complexity and preservation of file size for small hiding capacity, compared to previous methods. .

Keywords: JPEG images, reversible data hiding, adaptive hiding capacity, termination point, bitstreams

This research was supported by Natural Science Foundation of China (Grant 61372175) and National Key Laboratory Foundation of China (Grant 2018SSFNKLSMT-13). We express our thanks to Ms. Lee who checked our manuscript.

http://doi.org/10.3837/tiis.2020.01.022 **ISSN : 1976-7277**

1. Introduction

With the popularization and the development of internet, a huge amount of data is generated and transmitted, and the protection of privacy attracts accumulating focus. Data hiding is a technology that embeds secret data in host images and has extensive application. In fields e.g. computed tomography (CT) reports, military target and legal instrument, additional information needs to be transmitted together, but there is a frequent demand to preserve the integrity of original images. Thus Reversible data hiding (RDH) [1] is proposed, which can reconstruct both host images and additional data with little distortion. Generally, RDH can be divided into four categories [2]: space domain [3-5], transform domain [6][7], compress domain [8-11] and encrypt domain [12-14]. Only RDH in compress domain can be used in compressed image.

For the advantages of low complexity and high efficiency, Joint Photographic Experts Group (JPEG) is used as a major image compression technique [15][16]. Four kinds of RDH methods are established independently in JPEG images, VLC mapping [17-20], quantified DCT coefficients modification [21-25] and concealment in encrypted JPEG bitstreams [26][27].

Mobasseri et al. [17] proposed the first kind of VLC mapping scheme to flip appended bits or a bit of variable-length-code (VLC), but in the problem that decoding failure is unresolved. Qian and Zhang et al. [18] proposed the mapping algorithm to relate used VLCs to unused VLCs, and modified the Huffman Table in JPEG header. But only a fraction of the total 162 VLCs in Huffman encoding table are used, and then some unused VLCs are mapped to used VLCs to represent secret data. However, pairs of VLCs are not in the same category, so mapped frequencies might exceed 64. Hu et al. [19] improved Qian and Zhang et al.'s [18] method. In the improved method, mapping is sorted in occurrence frequencies, so in a category the most used VLC is mapped to several unused VLCs, while others are mapped to only one unused VLC. Qiu et al. [20] proposed a RDH scheme in JPEG images, and established an alternative embedding algorithm using code mapping and reordering. In such method, different combination of unused VLCs and used VLCs remapping will increase hiding capacity, but marked images can be recovered only after secret data is fully extracted. In addition, marked images cannot be displayed normally in VLC mappings.

Wang et al. [21] first proposed a method to display marked images in high PSNR and obtain large hiding capacity. Both discrete cosine transform (DCT) coefficients and quantization table are modified so that file expansion is suppressed slightly. But it is not suitable for images with Quality Factor (QF) of 100. Huang et al. [22] proposed a histogram shifts (HS) scheme that selects coefficients in each DCT block with values -1 and 1 to embed secret data, and achieved a further improvement in capacity by statistics of values 0 in DCT blocks. However, file expansion is not appropriately addressed. Qian et al. [23] proposed a novel algorithm with improved capacity that uses appropriate blocks and coefficients for data embedment in HS order. However, there is little improvement regarding file expansion and image distortion. Hou et al. [24] optimized Huang et al.'s [22] method to extract K frequencies. In such method, there is the least deviation in DCT blocks, and coefficients in {-1, 1} are shifted at the K frequencies. It has better image quality and less file expansion with the same hiding capacity. Liu et al. [25] proposed a simple and efficient scheme that all nonzero coefficients in each DCT block are modified to hide secret data. Hiding capacity is substantially enhanced and file expansion is well preserved.

In order to realize transmission security, Qian et al. [26] proposed the first JPEG encryption algorithm, which complies with JPEG decoding criterion, and embeds an additional message into the encrypted bitstreams without changing file size. Original bitstreams can be reconstructed intactly using an iterative recovery algorithm based on the blocking artefact at the expense of a large computational amount. Chang et al. [27] proposed a hiding method in encrypted JPEG bitstreams, in which a part of original JPEG bitstreams are recompressed to reserve spare space to embed secret data, and recompressed file is encrypted and synthesized with secret data in the end. Privacy security is highly guaranteed, but marked images cannot be displayed normally without decryption and images can be only recovered after secret data extraction. These methods also need additional information to decrypt bitstreams, and blind extraction of secret data is not feasible.

Although these RDH methods can perfectly embed secret data in JPEG images, file expansion and invisibility of marked images are not yet solved, and image recovery also depends on secret extraction. In this paper, we propose a RDH method in JPEG bitstreams that the least important parts are removed to reserve spare space. File size does not increase after the concealment, and secret data is embedded as comment part following marker APPn in file header similar to Richter et al.'s [28] and Zhang et al.'s [29] method. The first substituted frequencies in each 8×8 block are almost the same and embedding is in accordance with JPEG encoding criterion for a normal display of the marked image. With small amount of secret data, marked image quality is perfect such that the concealment can be taken as almost lossless. The objectives of this paper includes: (1) unchanged JPEG file size; (2) normal display of marked images; (3) adaptive and large hiding capacity.

The rest of the paper is organized as follows. Previous JPEG RDH algorithms are reviewed in Section 2. In Section 3, details of the proposed RDH method including steps of hiding, extracting and recovering are provided. Section 4 shows and explains the experimental results. Finally advantages and disadvantages of the scheme are summarized in Section 5.

2. Related Work

Since the main concern is paid to no file expansion and normal display of marked images, then two classical methods will be introduced briefly.

2.1 Qian et al.'s [26] method

Qian et al. [26] constructed a RDH framework shown in **Fig. 1**. Before data hiding, JPEG file is decompressed to obtain 8×8 DCT blocks and corresponding bitstream, and K blocks are pseudo-randomly selected from the whole N blocks. Subsequently, N-K blocks are recompressed and comment part is inserted following marker APPn in file header, while K blocks are encrypted and reshaped to construct a new JPEG file. Pre-processed JPEG images can still be displayed but are completely different from original images. Then bitstreams of N-K blocks are parsed so that codes of DC and AC coefficients are clustered into a set, while appended bits are taken as another set. Appended bits are separately divided into several groups in which bits size is the same, then with a compression matrix H, they are compressed to reserve spare space so that secret data could be inserted in the end. In receiver, with decryption key and transpose matrix G, secret data can be directly losslessly extracted, while compressed appended bits need many iterations to search the optimal coefficients. Then with the least deviation to approach blocks, images can be almost losslessly recovered except few candidate blocks, because some information is missing in the compression. After decryption, marked images, which contain secret data, can still be displayed normally but visual quality is

just acceptable by human eyes. Some side information is needed to find compressed file size and used for decryption of retained bitstreams.

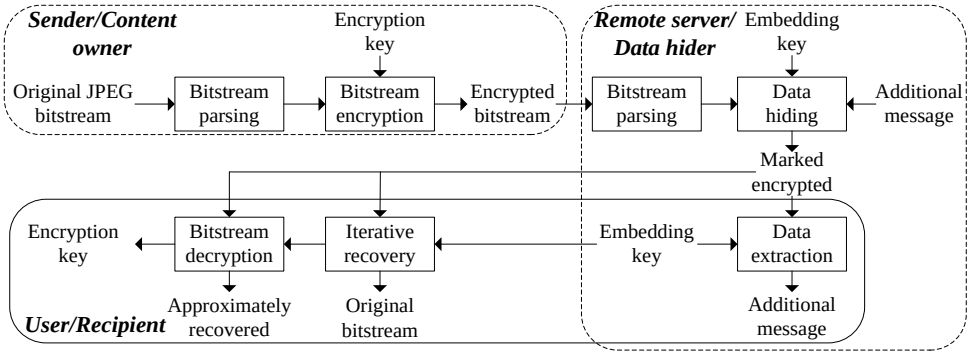


Fig. 1. Framework of Qian et al.'s [26] method

2.2 Chang et al.'s [27] method

Chang et al. [27] proposed a RDH method in encrypted JPEG bitstreams, in which marked images can be displayed with a high PSNR value after decryption. He adopted the framework of the reserving-room-before-encryption in Ma et al.'s [30] work, which first compressed bitstream before encryption, as shown in **Fig. 2**. Quantified DCT coefficients which are not inside {-1, 0, 1} are selected and compressed in his proposed approach. Bitstreams corresponding to these two least planes are lossless compressed so that some spare space is reserved to embed secret data. Then secret data are transmitted into bitstreams and filled in the end so that synthetic bitstreams are used as recoded data of these two least planes. Finally the separate encryption for data part and header can achieve privacy protection in concealment. In receiver, marked images can be displayed normally after decryption and because modification just mostly ranges in {-3, -2, 2, 3}, deviation between host JPEG images and decrypted marked images is ignorable such that it cannot be discerned by human eyes. With side information which refers to compressed file size, secret data can be extracted and host images can also be recovered lossless.

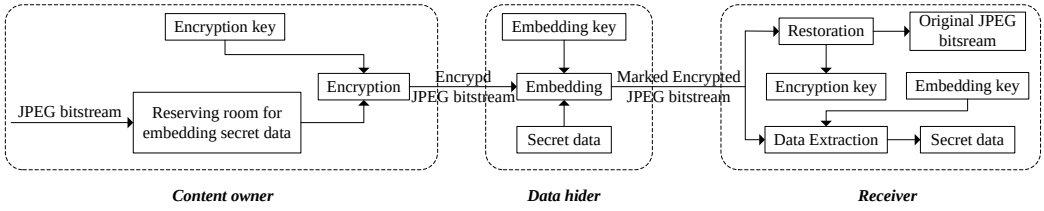


Fig. 2. Framework of Chang et al.'s [27] reserving-room-before-encryption

3. Proposed method

3.1 JPEG encoding criterion

JPEG compression technique transformed data from the space domain to DCT domain to exploit the redundancy because the relevance in the DCT domain is stronger than that in space

domain. The compression procedure consists of five steps in general, as is shown in **Fig. 3**. The decompression is the opposite.

Fig. 3. JPEG compression procedure

After the quantization of the DCT transform, data are concentrated in the low frequencies, while they are sparsely distributed in the high frequencies. To generate consecutive zero coefficients for convenient encoding, the low frequency components appear first and the high frequency components follow, such that the whole coefficients are arranged in "Zigzag" mode, as shown in **Fig. 4**. There are totally 64 frequencies in a block. The most important frequencies, including direct current (DC) and alternating current (AC) components, are located at the left and top corner, while the least important frequencies are at the right and bottom corner.

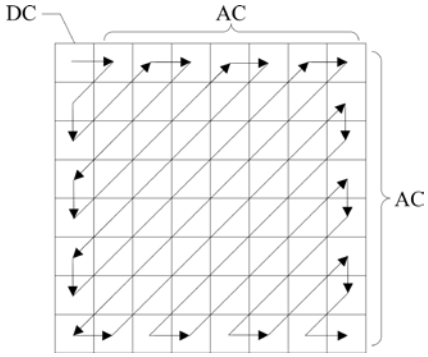


Fig. 4. The sketch of the "Zigzag" scan

3.2 Adaptive search for termination point

In each quantified 8×8 DCT block, 64 coefficients are scanned in "Zigzag" order and are encoded in Huffman codes, as shown in **Table 1**. Encoded data is arranged in *RSV* (*Run*/*Size* and *Value*) order, where *Run* refers to zero numbers before nonzero, *Size* refers to nonzero bits size, and *Value* refers to appended nonzero coefficient. There are some zeros interval between nonzero coefficients, then these zeros and nonzero are jointly encoded such that *code length* refers to *Run*/*Size* bits size, *codes* refers to *Run*/*Size* binary code, *sum length* refers to the total binary size of *RSV*, *Rate* refers to the binary size proportion of nonzero codes in *Run*/*Size* and end of bits (EOB) '1010' denotes encoding finished in a block. For example, in a small bitstream '11000', *Run*/*Size* '1100' represents that the number of zeros before nonzero is one, nonzero coefficient is -1 and then it can be taken as *RSV* form (1, -1).

Information is concentrated in low frequencies where distribution of nonzero coefficients is dense with large absolute values, while in high frequencies distribution is sparse with extensive presence of zeros. Besides, *sum length* expands and *Rate* reduces obviously when *Run* is larger than 4 and *Size* is smaller than 3. Then we can choose a suitable high frequency as the termination point. The nonzero coefficients for frequency above the termination point are reset to zero, and corresponding bitstreams are removed so that the least important information is missing.

JEPG compression is featured with an insufficient robustness and the deficiency to correct errors, such that a bit of flipping would cause decoding failure. If we directly remove a fraction of bitstreams, no termination point and EOB can be ascertained and thus start bits index is

missing in the next block. Then we need to have a judgement whether a block can be compressed, adjustment of termination point in each block and some supplementary information.

Table 1. AC coefficients in JPEG Huffman codes

Run/Size	*Value*	*code length*	*codes*	*sum length*	*Rate*
0/0(EOB)	0	0	1010	4	0
0/1	-1,1	2	00	3	0.5
0/2	-3,-2,2,3	2	01	4	1
0/3	-7,-6,-5,-4,4,5,6,7	3	100	6	1
0/4	-15,-14,…-9,-8,8,9,…14,15	4	1011	8	1
0/5	-31,-30,…-17,-16,16,17,…30,31	5	11010	10	1
1/1	-1,1	4	1100	5	0.25
1/2	-3,-2,2,3	5	11011	7	0.4
1/3	-7,-6,-5,-4,4,5,6,7	7	1111001	10	0.43
1/4	-15,-14,…-9,-8,8,9,…14,15	9	111110110	13	0.44
2/1	-1,1	5	11100	6	0.2
2/2	-3,-2,2,3	8	11111001	10	0.25
2/3	-7,-6,-5,-4,4,5,6,7	10	1111110111	13	0.3
3/1	-1,1	6	111010	7	0.17
3/2	-3,-2,2,3	9	111110111	11	0.22
4/1	-1,1	6	111011	7	0.17
4/2	-3,-2,2,3	10	1111111000	12	0.2
5/1	-1,1	7	1111010	8	0.14
5/2	-3,-2,2,3	11	11111110111	13	0.18
6/1	-1,1	7	1111011	8	0.14
6/2	-3,-2,2,3	12	111111110110	14	0.17
…	…	…	…	…	…

There are 63 candidate points at most from the 1st to the 63th frequency in the whole image, and if we choose an arbitrary frequency *T* as the termination point, there are five cases as listed in **Fig. 5** '*' denotes an arbitrary nonzero value. If the last nonzero coefficient occurs with frequency below *T* in case 1, encoding is finished. In case 2, the coefficient in point *T* is nonzero, but no nonzero values exist above *T*. Similar with case 2, the coefficient in point *T* is zero and there exists a nonzero value above *T* in case 3. These three cases are unavailable for our method because no spare bits can be reserved, and encoding is finished with EOB '1010'. Then following two cases, where there are more than one nonzero with frequencies above *T*-1, can be used for coefficients compression. In case 4, the coefficient in point *T* is nonzero and there are some nonzero coefficients above *T*. Thus we can take *T* as the termination point and remove bitstreams above *T* and add end marker '1010'. It is ideal condition that the termination point in this block is equal to *T*. However, the first nonzero frequency above *T*-1 is more likely to be larger than *T*, as illustrated by case 5. Accordingly the first occurrence frequency *T'* is used as the termination point, and end marker '1010' is added. In such a way, no decoding failure occurs and when it is zero in point *T*, an adaptive scheme would be incorporated and allows a joint encoding for the following nonzero frequency.

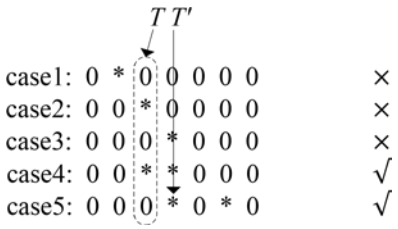


Fig. 5. Five cases in termination point adjustment

We construct a spare space mapping between reserved spare space and termination point in each 8×8 DCT block, and then calculate the summary to build the model in an image. Hiding capacity in the 64th frequency is zero and we set 63 as the highest point. Given that hiding capacity is M bits, we can find a frequency T with which spare space is smaller than M while with point T-1 it is larger than M. If we take frequency T-1 as termination point, some blocks will be unused and in the truncated blocks, leading to deteriorated information loss. Then we can choose frequency T as termination point in an image and further compress data with a function to ensure that each block can be substituted.

$$\sum_{i=1}^{K} Capacity_i^T + \sum_{j=K+1}^{N} Capacity_j^{T-1} = M \tag{1}$$

Where, N is blocks number in an image, K is divided blocks index that in the first K blocks termination point is T and in the following N-K blocks termination point is T-1, $Capacity_i^T$ refers to spare space in the i^{th} block when T is termination point and M is the given hiding capacity.

Then we can adaptive adjust our termination point and blocks according to specified capacity so that distortion in each block is almost the same and optimal visual quality can be achieved.

3.3 Embed secret data in file header

A comment marker APPn is used to insert in file header, and secret data is put behind it.

To protect the privacy of secret data, some pre-processing measures are assumed to encrypt it. The first step is to generate a random seed S and corresponding random sequence Seq in the same size with secret bitstream Ori, then secret data is encrypted with NOR operation.

$$En(m) = NOR(Ori(m), Seq(m)), m = \{1, 2, 3, ...N-1, N\} \tag{2}$$

where, m is bit index, N is bits size of secret data, $Ori(m)$ is secret bit in index m, $Seq(m)$ is random bit in index m, $En(m)$ is encrypted bit in index m.

With four appended bytes ahead, constitution of additional data is shown in **Fig. 6**.

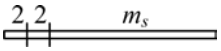


Fig. 6. Additional data constitute

The spare space M is equal to 2+2+m_s, and the first two bytes are comment marker FFFE, and the following two bytes are the byte size of additional data, then the rest are encrypted secret data. The concealment is consistent with JPEG encoding criterion so that the marked JPEG file can be normally decompressed by any receiver.

3.4 An illustration for truncating DCT coefficients and bitstreams

Supposing that an 8×40 8-bits grayscale JPEG image is decompressed to quantified DCT domain, then five 8×8 blocks are obtained as shown in **Fig. 7**.

10	-9	0	-2	0	0	0	0
6	-3	0	0	0	0	0	0
4	-1	1	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

a

9	-7	4	0	-1	0	0	0
8	4	3	0	0	0	0	0
-6	0	-2	0	0	0	0	0
0	-1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

b

12	-4	2	0	0	0	0	0
3	5	0	-1	0	0	0	0
-4	0	0	0	0	0	0	0
1	0	2	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

c

-9	-6	2	5	-1	0	-1	0
4	3	-2	0	0	0	0	0
-4	0	0	0	0	0	0	0
-2	0	-1	0	0	0	0	0
1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

d

8	-6	4	2	0	0	0	0
5	6	0	-1	0	0	-2	0
-5	-1	-1	2	0	0	0	0
0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	0
0	-1	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0

e

Fig. 7. An original 8×40 DCT block

In **Fig. 7(*a*)**, the bitstreams are:
1011010,10110001,100110,100100,0101,1101100,11000,1110101,1010.
In **Fig. 7(*b*)**, the bitstreams are:
1011001,100011,10111000,100010,100100,100100,1101111,1110100,0100,11000,1010.
In **Fig. 7(*c*)**, the bitstreams are:
1011100,100000,0111,100000,100101,0110,1110101,1110100,111111100010,1010.
In **Fig. 7(*d*)**, the bitstreams are:
1011001,100010,100100,100000,0111,0110,100101,0100,1101100,001,1110100,1110100 ,1110101, 1110110,1010.
In **Fig. 7(*e*)**, the bitstreams are:
1011000,100010,100101,100001,100110,100100,0110,11000,11001,11000,000,11111011 110,1110110,11001, 111111100000,1110111,1010.

We select the 15th frequency as termination point, and coefficients below the point are retained but coefficients above are removed to reserve spare space. However, the 15th frequency is the theoretical value and the actual value display a significant variation according to the adaptive scheme as illustated by the five cases. We can obtain termination point in frequency map shown in **Fig. 8**.

1	2	6	7	15	16	28	29
3	5	8	14	17	27	30	43
4	9	13	18	26	31	42	44
10	12	19	25	32	41	45	54
11	20	24	33	40	46	53	55
21	23	34	39	47	52	56	61
22	35	38	48	51	57	60	62
36	37	49	50	58	59	63	64

a

1	2	6	7	15	16	28	29
3	5	8	14	17	27	30	43
4	9	13	18	26	31	42	44
10	12	19	25	32	41	45	54
11	20	24	33	40	46	53	55
21	23	34	39	47	52	56	61
22	35	38	48	51	57	60	62
36	37	49	50	58	59	63	64

b

1	2	6	7	15	16	28	29
3	5	8	14	17	27	30	43
4	9	13	18	26	31	42	44
10	12	19	25	32	41	45	54
11	20	24	33	40	46	53	55
21	23	34	39	47	52	56	61
22	35	38	48	51	57	60	62
36	37	49	50	58	59	63	64

c

1	2	6	7	15	16	28	29
3	5	8	14	17	27	30	43
4	9	13	18	26	31	42	44
10	12	19	25	32	41	45	54
11	20	24	33	40	46	53	55
21	23	34	39	47	52	56	61
22	35	38	48	51	57	60	62
36	37	49	50	58	59	63	64

d

1	2	6	7	15	16	28	29
3	5	8	14	17	27	30	43
4	9	13	18	26	31	42	44
10	12	19	25	32	41	45	54
11	20	24	33	40	46	53	55
21	23	34	39	47	52	56	61
22	35	38	48	51	57	60	62
36	37	49	50	58	59	63	64

e

Fig. 8. Termination point in frequencies map

In the first block shown in **Fig. 7(*a*)**, the encoding is finished at the 13th frequency such that no spare space is reserved. The block corresponds to case 1 and the actual value is 13th as shown in **Fig. 8(*a*)**.

In the second block shown in **Fig. 7(*b*)**, the encoding is just finished at the 15th frequency such that no spare space is reserved. The block corresponds to case 2 and the actual value is the 15th as shown in **Fig. 8(*b*)**.

In the third block shown in **Fig. 7(*c*)**, the encoding is finished at the 19th frequency and the coefficients at the 15th,16th,17th,18th frequencies are zero such that no spare space is also reserved. The block corresponds to case 3 and the actual value is the 19th as shown in **Fig. 8(*c*)**.

In the fourth block shown in **Fig. 7(*d*)**, the encoding is finished at the 34th frequency and the value at the 15th frequency is -1 such that there exists spare space. The block corresponds to case 4 and the actual value is the 15th shown in **Fig. 8(*d*)**.

In the fifth block shown in **Fig. 7(*e*)**, the encoding is finished at the 35th frequency and the value at the 15th frequency is 0 and the coefficients at the 23th,25th,30th,35th frequency are nonzero such that there exists spare space. The block corresponds to case 5 and the actual value is the 18th as shown in **Fig. 8(*e*)**.

Then we can obtain retained DCT blocks as shown in **Fig. 9**. In each block, the coefficients above the termination point are zero and corresponding bitstreams are replaced with end marker '1100'.

10	-9	0	-2	0	0	0	0
6	-3	0	0	0	0	0	0
4	-1	1	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

a

9	-7	4	0	-1	0	0	0
8	4	3	0	0	0	0	0
-6	0	-2	0	0	0	0	0
0	-1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

b

12	-4	2	0	0	0	0	0
3	5	0	-1	0	0	0	0
-4	0	0	0	0	0	0	0
1	0	2	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

c

-9	-6	2	5	-1	0	0	0
4	3	-2	0	0	0	0	0
-4	0	0	0	0	0	0	0
-2	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

d

8	-6	4	2	0	0	0	0
5	6	0	-1	0	0	0	0
-5	-1	-1	2	0	0	0	0
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

e

Fig. 9. An Retained 8×40 DCT block

In **Fig. 9(*a*)**, the retained bitstreams are:
1011010,10110001,100110,100100,0101,1101100,11000,1110101,1010.
In **Fig. 9(*b*)**, the retained bitstreams are:
1011001,100011,10111000,100010,100100,100100,1101111,1110100,0100,11000,1010.
In **Fig. 9(*c*)**, the retained bitstreams are:
1011100,100000,0111,100000,100101,0110,1110101,1110100,111111100010,1010.
In **Fig. 9(*d*)**, the retained bitstreams are:
1011001,100010,100100,100000,0111,0110,100101,0100,1101100,001,1110100,1010.
In **Fig. 9(*e*)**, the retained bitstreams are:
1011000,100010,100101,100001,100110,100100,0110,11000,11001,11000,000,11111011 110,1010.

Then spare bitstreams are 1110100,1110101,1110110,1110110,11001,111111100000, 1110111, such that 52 bits can be removed as spare space in the small image.

3.5 Procedure of data hiding and extraction

In concealment, it is of the most importance to decompress DCT coefficients, which includes the following steps:

Decompress JPEG file to DCT blocks and divide corresponding bitstreams.

Truncate coefficients in each block and construct a spare space mapping in an image.

With a random seed, we encrypt secret data with random bitstreams.

According to the given hiding capacity, we can calculate termination point T, with which spare space can provide enough space for secret data.

To ensure that each block is used, we further search for an optimal block index with a termination point of T in the first fraction and termination point of T-1 in latter fraction.

Insert comment marker APPn and encrypted secret data in file header.

Synthesize file header and compressed data body to rebuild a marked JPEG file.

The concealment procedure is shown in **Fig. 10**.

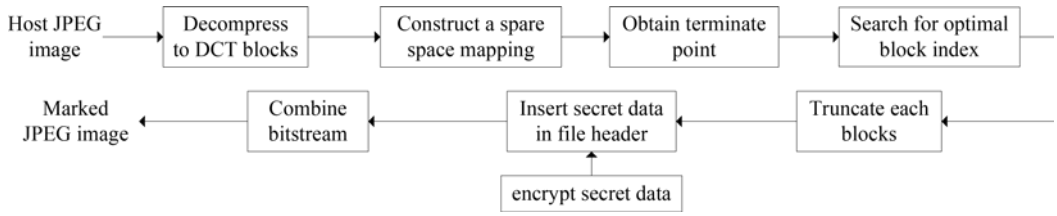


Fig. 10. Framework of concealment in proposed method

Secret data is directly extracted following APPn and a receiver with the permission to own seed can generate the same binary random sequence. Secret data can be further decrypted lossless with *NOR* operation. Without post-processing, it is only to remove secret data in marked file header, and the shortened file is taken as the final recovered file.

4. Experimental results

To evaluate the performance of the proposed work, we take eight standard images in library [31] separately compressed with QF 20,50,80 as host JPEG images, and secret data from random bitstreams with size 500,1000:1000:25000 bits are used to analyse the influence of hiding capacity in host images. Images Lena, Airplane, Baboon, Boat, Bridge, Couple, Elaine, Peppers are in size of 512×512 so that 4096 blocks can be decompressed.

PSNR is used to measure visual quality of marked JPEG images compared with host JPEG images. Because post-processing has no improvement to recovery, PSNR of marked images are the same with recovered images. Embedding capacity (EC) is secret data size which equals to additional size minus 32 appended bits. Also Time cost (TC) is to measure computational complexity of concealment.

The platform in the experiment is the win7 64 bits system, software is MATLAB 2013b, and CPU is the e3v3 Xeon processor with four cores of 3.3GHz.

Concealment of image Lena, Peppers and Baboon with QF 80 is shown in **Fig. 11. Fig. 11**(*a*) shows host JPEG Lena, **Fig. 11**(*b*-*c*) show marked Lena with EC of 500 and 25000 bits respectively; **Fig. 11**(*d*) shows host JPEG Peppers, **Fig. 11**(*e*-*f*) show marked Peppers with EC of 500 and 25000 bits respectively; **Fig. 11**(*g*) shows host JPEG Baboon, **Fig. 11**(*h*-*i*) show marked Baboon with EC of 500 and 25000 bits respectively. There is almost no deviation between host JPEG images and marked images when EC is small. With EC up to 25000 bits, which accounts for 15% of the host JPEG, marked images can still be displayed normally and the distortion is difficult to distinguish with human eyes.

Fig. 11. Three tested images in concealment

PSNR values are listed in Table 2 for QF of 20,50,80 and EC of 500,25000 bits, as shown in **Table 2**. With EC=500 bits and QF=80, PSNR of marked images is mostly larger than 50 dB, and with QF=20, PSNR is still larger than 40 dB. With EC=25000 bits and QF=80, PSNR of marked images is mostly larger than 35 dB, and they are around 30 dB with QF=20. In the worst condition, visual quality of marked images is just acceptable and basic information is retained.

Table 2. PSNR versus different QF and EC (:dB)

QF	EC	Lena	Airplane	Baboon	Boat	Bridge	Couple	Elaine	Peppers
20	500	45.21	41.96	39.77	42.35	42.57	43.45	46.80	43.63
	25000	29.40	29.28	26.76	29.14	29.35	29.62	30.91	28.90
50	500	47.41	47.29	42.85	46.61	46.65	45.70	49.52	46.95
	25000	33.63	33.84	28.90	32.54	31.45	32.85	36.98	33.78
80	500	53.02	53.48	47.96	51.92	49.29	50.90	50.96	50.99
	25000	37.93	37.95	33.14	37.06	34.86	35.77	38.73	37.21

To compare EC and PSNR, experiments from two other methods are provided in **Table 3**. In Chang et al.'s [27] RDH-EI method spare space is reserved by encryption and the

information corresponding to the least two bit planes is recompressed losslessly. Without encryption, marked images can be decoded directly by any receiver, because the concealment follows JPEG encoding criterion. In Qian et al.'s [26] method, some arbitrary blocks are selected as side information and AC coefficients in other blocks are compressed to reserve spare space. With a great number of iterations to search for optimal parameters, images can be recovered almost lossless. Both of the two methods compress data before data hiding, but considerable modification in AC coefficients is involved. Chang et al. [27] modifies the least two planes of AC coefficients, while Qian et al. [26] modifies the whole fraction of AC coefficients in unselected blocks. Then PSNR and EC in Chang et al.'s [27] method are both larger than those in Qian et al.'s [26] method with the same QF. Similar with these methods, we compress AC coefficients before data hiding. We have an estimation of data importance in DCT domain such that coefficients with high frequencies and small absolute value represent little importance, and then we can remove these corresponding bitstreams to reserve spare space. With the proposed function, an optimal termination point and divided blocks index can be resolved such that deviation is reduced as much as possible for the same EC. For privacy protection, secret data is pre-processed with encryption and inserted as comment part following marker APPn in file header. Consequently, marked images can be displayed with a high PSNR without any post-processing. Experimental results show that for the same QF, both PSNR and EC in marked JPEG images are obviously improved, compared with previous works. The only disavantage is that we cannot recover lossless images, because original data corresponding to high frequencies are missing during compression.

Table 3. PSNR and EC comparison with Qian et al. [26] and Chang et al. [27]

Image	Method	QF=20		QF=50		QF=80	
		EC	PSNR	EC	PSNR	EC	PSNR
Lena	Proposed	1000	42.19	1000	44.72	2000	47.16
	Qian [26]	816	34.13	1364	37.46	1260	35.84
	Chang[27]	322	40.94	528	44.29	798	47.19
Airplane	Proposed	2000	38.05	2000	42.62	2000	47.81
	Qian [26]	739	33.28	1024	35.64	1320	36.59
	Chang[27]	527	38.12	1179	42.58	1069	47.69
Baboon	Proposed	2000	35.14	3000	36.83	6000	38.84
	Qian [26]	436	32.68	768	34.12	842	36.14
	Chang[27]	1207	35.16	1233	36.97	1555	38.78
Boat	Proposed	2000	37.91	2000	41.93	2000	44.59
	Qian [26]	625	36.64	726	37.69	826	38.42
	Chang[27]	409	38.38	722	41.63	1032	44.88
Bridge	Proposed	2000	37.31	3000	39.10	2000	44.09
	Qian [26]	964	37.91	1125	38.46	1064	39.74
	Chang[27]	374	37.42	510	39.26	634	43.95
Couple	Proposed	2000	38.52	2000	41.11	3000	43.93
	Qian [26]	578	32.68	769	34.16	862	35.21
	Chang[27]	445	38.91	747	41.16	1088	44.29
Elaine	Proposed	3000	40.44	3000	44.26	3000	45.99
	Qian [26]	869	36.52	964	37.19	1038	38.22
	Chang[27]	401	40.68	624	44.35	968	45.81
Peppers	Proposed	1000	40.67	1000	44.41	2000	47.27
	Qian [26]	496	36.82	762	37.24	846	39.16
	Chang[27]	291	40.61	450	43.98	960	47.35

When secret data increases from 500 to 25000 bits, PSNR is shown in **Fig. 12**. **Fig. 12(a)** shows PSNR values for QF=20, **Fig. 12(b)** shows PSNR values with QF=50, **Fig. 12(c)** shows PSNR values with QF=80. QF has a great influence in information entropy of host JPEG images. When QF=80, quantitation step is small and many nonzero coefficients are retained so that file size expands seriously. When QF=20, large quantitation step and small coefficients result in little change of file size. With the same EC, PSNR of marked images with higher QF is larger than that with lower QF, because relative distortion is smaller. With 1000 secret bits added, average PSNR of marked images can be reduced by 0.5 dB. When secret size is small, PSNR of marked images are sensitive to EC. However, when secret size is large, PSNR shows less dependence on EC. When QF is 50, PSNR is almost proportional to marked file size.

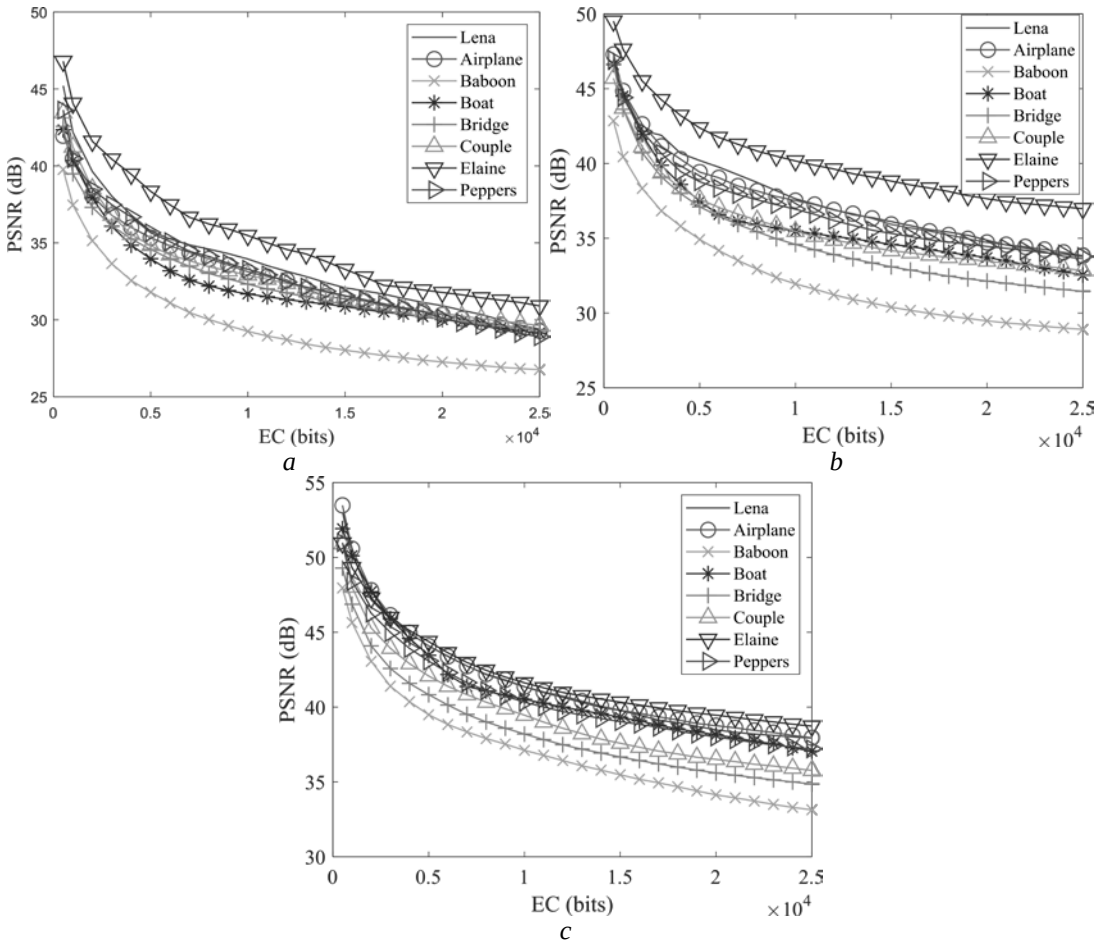


Fig. 12. PSNR versus EC with different QF

We show termination points with different QF and EC in **Fig. 13**. **Fig. 13(a)** shows termination point with QF 20, **Fig. 13(b)** shows termination point with QF 50, **Fig. 13(c)** shows termination point with QF 80. Termination point represents visual quality, since higher termination point indicates less loss of original entropy and larger PSNR of marked images. Image Baboon is more complex than image Lena that information of Baboon is less concentrated in low frequencies and termination point is higher. Although termination point of Baboon is higher, average of coefficients is smaller and more nonzero coefficients exist above the termination point, indicating that more original information is missing, thus PSNR is

smaller. Generally, when termination point is larger than 35, retained information images can be recovered with PSNR above 38 dB so that visual quality is quite acceptable by human eyes.

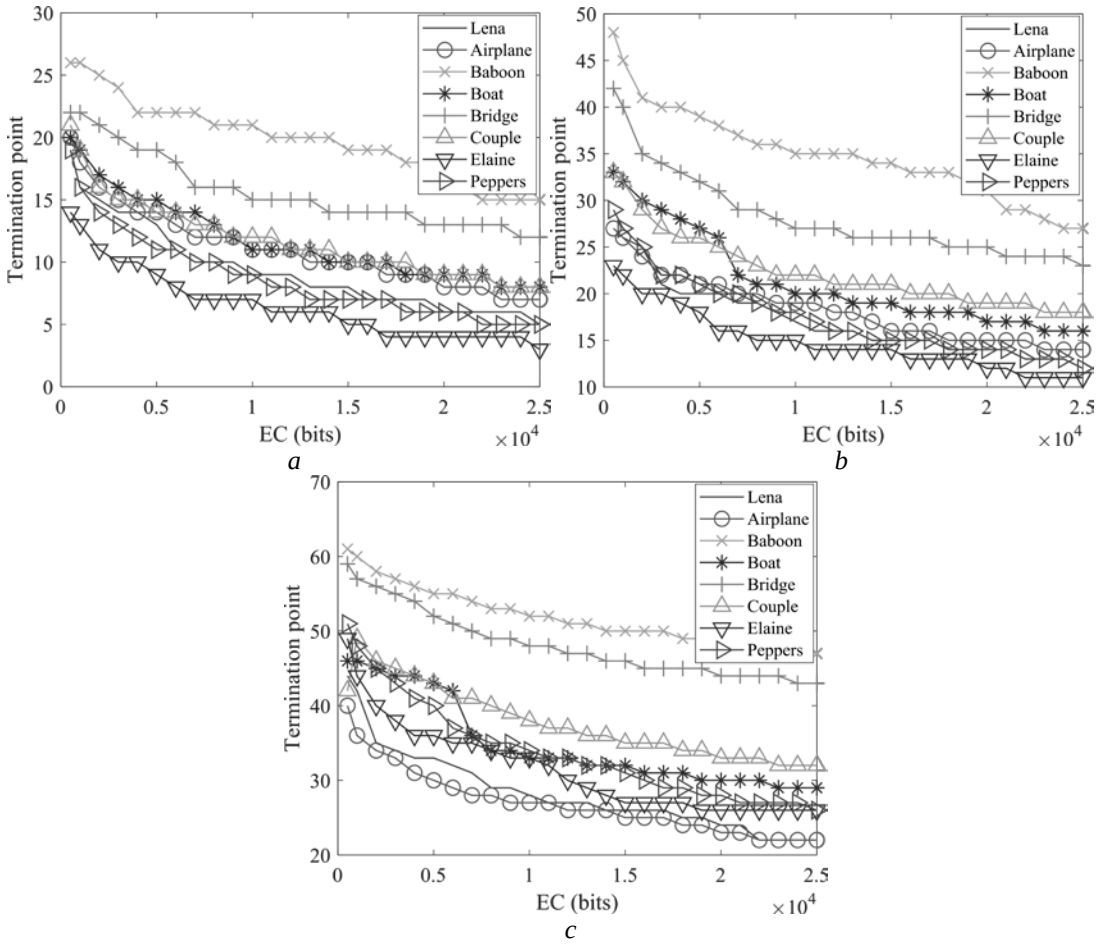


Fig. 13. Termination point versus EC with different QF

We also make a comparison of TC with previous algorithms, as shown in **Fig. 14**. We take 3000 bits as EC that marked images can be displayed in a high quality and EC in Qian et al. [26] and Chang et al. [27] are variable and mostly less than 1500 bits. TC in three methods are not varied largely and is less than 1 second. Besides, concealment can be real-time processed and complexity is not considerable. TC in our method exhibits a smallest value because the construction of spare space mapping in DCT blocks is the main work and it can be obtained from side information during bitstreams decoding and it is quite different from the other two methods. With deletion of the least important bitstreams and adding of little appended information, it can provide enough spare space to embed secret data directly in file header, and little time is needed to extract secret and recover images. While extra search is needed in previous works before embedding secret, both secret data and images can only be recovered with post-processing.

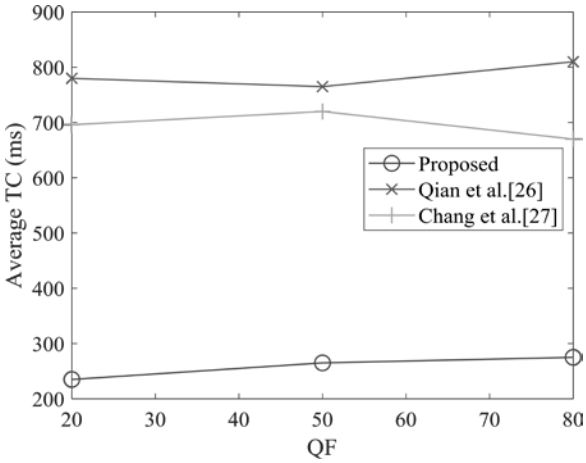


Fig. 14. Average TC comparison in different methods

As a supplement of the experiment, we also test images on the standard Kodak dataset [32]. There are 24 raw gray images with size of 3076×2048, and we compress them with QF 20,50,80 as host JPEG images. In methods of Qian et al. [26] and Chang et al. [27], EC in each tested image is not the same, such that we take the maximum of EC in two methods as EC_{max} and take two times of EC_{max} as our EC in each compressed image. The average EC of the proposed method in QF 20,50,80 is shown in **Table 4**.

Table 4. The relation of average EC with QF

QF	20	50	80
Average EC	33120	45024	48624

The average PSNR of marked JPEG images comparison is shown in **Fig. 15.** It shows that the proposed method produces the largest EC as well as the highest PSNR, indicating the best image quality. The concealment in the JPEG image Motorbike with QF 50 is shown in **Fig. 16.** **Fig. 16(*a*)** is the retained image, and **Fig. 16(*b*)** is the host image. The EC is 52016 bits, however, termination point is 12, PSNR is 47.37dB, and the distortion cannot be distinguished by human eyes such that the concealment can be considered almost lossless.

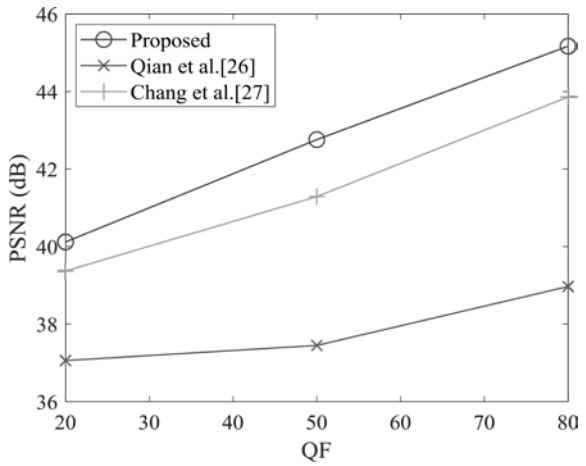


Fig. 15. Average PSNR comparison in different methods from the Kadak dataset [32]

a *b*

Fig. 16. Concealment in JPEG image Motorbike from the Kadak dataset [32]

5. Conclusion

We propose a novel RDH technique in JPEG bitstreams. A creatively space mapping is put forward, in which the least important bitstreams are found and removed. Data truncation complies with JPEG codes, and secret data are inserted in header as comment part so that file size does not expand. Besides, it makes a good balance between visual quality and hiding capacity. When hiding capacity is small, marked images can be displayed with PSNR greater than 45 dB, so that the concealment can be considered almost lossless. And when EC is up to 25000 bits, marked images can still be displayed acceptable in receiver with a PSNR around 30 dB. Hiding capacity is variable and marked compressed files can be decoded normally in any receiver, but secret data cannot be decrypted correctly without random seed, thus privacy is highly protected. There is low complexity for extraction and recovery, and the whole concealment takes a short time. Our method outperforms previous works in terms of PSNR of marked images, hiding capacity and time cost.

References

[1] A. Khan, A. Siddiqa, S. Munib, and S. A. Malik, "A recent survey of reversible watermarking techniques," *Information sciences*, vol. 279, pp. 251-272, 2014. Article (CrossRef Link).

[2] M. Chandan, M. Pandey, "A Review on current Methods and application of Digital image Steganography," *International Journal of Multidisciplinary Approach & Studies*, vol. 2, no. 2, pp.163-170, 2015.

[3] J. Tian, "Reversible data embedding using a difference expansion," *IEEE transactions on circuits and systems for video technology*, vol. 13, no. 8, pp. 890-896, 2003. Article (CrossRef Link).

[4] D. M. Thodi, J. J. Rodríguez, "Expansion embedding techniques for reversible watermarking," *IEEE transactions on image processing*, vol. 16, no. 3, pp. 721-730, 2007. Article (CrossRef Link).

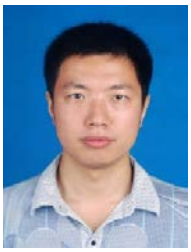
[5] Z. C. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," *IEEE Transactions on Circuits Systems Video Technology*, vol. 16, no. 3, pp. 354-362, 2006. Article (CrossRef Link).

[6] S. Lee, C. D. Yoo, and T. Kalker, "Reversible image watermarking based on integer-to-integer wavelet transform," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 321-330, 2007. Article (CrossRef Link).

[7] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital watermarking and steganography, 2nd Edition*, Morgan Kaufmann, San Francisco, pp.204-206, 2007.

[8] C.C Chang, Y. P. Hsieh, and C. Y. Lin, "Lossless data embedding with high embedding capacity based on declustering for VQ-compressed codes," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 341-349, 2007. Article (CrossRef Link).
[9] A. Westfeld, "F5-a steganographic algorithm," in *Proc. of International workshop on information hiding, Springer, Berlin, Heidelberg*, pp. 289-302, April, 2001.
[10] N. Provos, "Defending Against Statistical Steganalysis," in *Proc. of the 10th USENIX Security Symposium, IEEE, Washington D.C., USA*, pp. 323-335, March, 2001.
[11] W. Hong, Y. B. Ma, H. C. Wu, and T. S. Chen, "An efficient reversible data hiding method for AMBTC compressed images," *Multimedia Tools and Applications*, vol. 76, no. 4, pp. 5441-5460, 2017. Article (CrossRef Link).
[12] Z. X. Qian, X. P. Zhang, "Reversible data hiding in encrypted images with distributed source encoding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 4, pp. 636-646, 2016. Article (CrossRef Link).
[13] X. Cao, L. Du, X. Wei, D. Meng, and X. Guo, "High capacity reversible data hiding in encrypted images by patch-level sparse representation," *IEEE transactions on cybernetics*, vol. 46, no. 5, pp. 1132-1143, 2016. Article (CrossRef Link).
[14] X. P. Zhang, J. Long, Z. Wang, and H. Cheng, "Lossless and reversible data hiding in encrypted images with public-key cryptography," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 9, pp.1622-1631, 2016. Article (CrossRef Link).
[15] Int. Telecommunication Union, CCITT Recommendation T.81, Information Technology-Digital Compression and Coding of Continuous-tone Still Images-Requirements and Guidelines, 1992.
[16] A. M. Raid, W. M. Khedr, M. A. El-Dosuky, and W. Ahmed, "Jpeg image compression using discrete cosine transform-A survey," *International Journal of Computer Science & Engineering Survey (IJCSES)*, Vol.5, No.2, pp. 39-47, April 2014. Article (CrossRef Link).
[17] B. G. Mobasseri, R. J. Berger, M. P. Marcinak, and Y. J. NaikRaikar, "Data embedding in JPEG bitstream by code mapping," *IEEE Transactions on Image Processing*, vol. 19, no. 4, pp. 958-966, 2010. Article (CrossRef Link).
[18] Z. X. Qian, X. P. Zhang, "Lossless data hiding in JPEG bitstream," *Journal of Systems and Software*, vol. 85, no. 2, pp. 309-313, 2012. Article (CrossRef Link).
[19] Y. J. Hu, K. Wang, and Z. M. Lu, "An improved VLC-based lossless data hiding scheme for JPEG images," *Journal of Systems and Software*, vol. 86, no. 8, pp. 2166-2173, 2013. Article (CrossRef Link).
[20] Y. Qiu, H. He, Z. Qian, S. Li, and X. Zhang, "Lossless data hiding in JPEG bitstream using alternative embedding," *Journal of Visual Communication and Image Representation*, vol. 52, pp. 86-91, 2018. Article (CrossRef Link).
[21] K. Wang, Z. M. Lu, and Y. J. Hu, "A high capacity lossless data hiding scheme for JPEG images," *Journal of systems and software*, vol. 86, no. 7, pp. 1965-1975, 2013. Article (CrossRef Link).
[22] F.J. Huang, X. C. Qu, H. J. Kim, and J.W. Huang, "Reversible data hiding in JPEG images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 9, pp. 1610-1621, 2016. Article (CrossRef Link).
[23] Z. Qian, S. Dai, and B. Chen, "Reversible Data Hiding in JPEG Images Using Ordered Embedding," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 2, pp. 945-958, 2017. Article (CrossRef Link).
[24] D. Hou, H. Wang, W. Zhang, and N. Yu, "Reversible data hiding in JPEG image based on DCT frequency and block selection," *Signal Processing*, vol. 148, pp. 41-47, 2018. Article (CrossRef Link).
[25] Y. J. Liu, C.C Chang, "Reversible data hiding for JPEG images employing all quantized nonzero AC coefficients," *Displays*, vol. 51, pp. 51-56, 2018. Article (CrossRef Link).
[26] Z. Qian, H. Zhou, X. Zhang, and W. Zhang, "Separable reversible data hiding in encrypted JPEG bitstreams," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 6, pp. 1055-1067, 2016. Article (CrossRef Link).

[27] J. C. Chang, Y. Z. Lu, and H. L. Wu, "A separable reversible data hiding scheme for encrypted JPEG bitstreams," *Signal Processing*, vol. 133, pp. 135-143, 2017. Article (CrossRef Link).
[28] T. Richter, A. Artusi, and T. Ebrahimi, "JPEG XT: A new family of JPEG backward-compatible standards," *IEEE Multimedia*, vol. 23, no. 3, pp. 80-88, 2016. Article (CrossRef Link).
[29] S. R. Zhang, Q. D. Li, and Q. Zhou, "A JPEG image information hiding method based on datastreams," *CN107071455A*, 2017.
[30] K. Ma, W. Zhang, X. Zhao, N. Yu and F. Li, "Reversible data hiding in encrypted images by reserving room before encryption," *IEEE Transactions on information forensics and security*, vol. 8, no. 3, pp. 553-562, 2013. Article (CrossRef Link).
[31] The USC-SIPI Image Database, [Online], http://sipi.usc.edu/database/, 2006.
[32] The standard Kodak dataset, [Online], http://www.math.purdue.edu/~lucier/PHOTO_CD/, 2017.



Mingming Zhang received the B.S. degree in simulation engineering from National University of Defense Technology, Changsha, P.R.China in 2011. He received the M.S. degree in communication engineering from China Academy of Space Technology, Xi'an, P.R.China in 2015. He is currently working toward the Ph.D. degree in aircraft designing at China Academy of Space Technology, Xi'an, P.R.China. His research interests include data hiding and image compression.

Quan Zhou received the B.S., M.S. and Ph.D. degree in communication engineering from XIDIAN University, Xi'an, P.R.China in 1986, 1989 and 1992 respectively. He is currently working as a full professor in National Key Laboratory of Science and Technology on Space Microwave at China Academy of Space Technology, Xi'an, P.R.China. His research interests include digital image processing, data hiding and image compression.

Yanlang Hu received the B.S. degree in electronic and information engineering from Hohai University, Nanjing, P.R.China in 2005. He received the M.S. degree in communication engineering from China Academy of Space Technology, Xi'an, P.R.China, in 2008. He is currently working as a senior engineer in National Key Laboratory of Science and Technology on Space Microwave at China Academy of Space Technology, Xi'an, P.R.China. His research interests include computer vision and data transmission.