Quantized DCT Coefficient Category Address Encryption for JPEG Image

Shanshan Li[1] and Yuanyuan Zhang[2]
[1] School of Information Engineering, Chang'an University
Xi'an, Shaanxi 710064 - China
[e-mail: sputnik@126.com]
[2] The First Affiliated Hospital of Xi'an Jiaotong University,
Xi'an, Shaanxi 710061, China
[e-mail: zhang.yuan.yuan@stu.xjtu.edu.cn]
*Corresponding author: Shanshan Li

Received August 21, 2015; revised December 8, 2015; revised January 14, 2016; accepted February 23, 2016; published April 30, 2016

Abstract

Digital image encryption is widely used for image data security. JPEG standard compresses image with great performance on reducing file size. Thus, to encrypt an image in JPEG format we should keep the quality of original image and reduced size. This paper proposes a JPEG image encryption scheme based on quantized DC and non-zero AC coefficients inner category scrambling. Instead of coefficient value encryption, the address of coefficient is encrypted to get the address of cipher text. Then 8*8 blocks are shuffled. Chaotic iteration is employed to generate chaotic sequences for address scrambling and block shuffling. Analysis of simulation shows the proposed scheme is resistant to common attacks. Moreover, the proposed method keeps the file size of the encrypted image in an acceptable range compared with the plain text. To enlarge the cipher text possible space and improve the resistance to sophisticated attacks, several additional procedures are further developed. Contrast experiments verify these procedures can refine the proposed scheme and achieve significant improvements.

Keywords: Image encryption, JPEG, DCT, chaotic sequence, file size

This research was supported by National Natural Science Foundation of China (Grant No. 61402051, 51278058) and the 111 Project (No.B14043).

http://dx.doi.org/10.3837/tiis.2016.04.018 **ISSN : 1976-7277**

1. Introduction

The continued growth of personal intelligent devices and Internet makes it easy to distribute, share and exchange digital image data through various sorts of open networks. But, the convenience offered by the advances in open network communications has caused a security threat for storage and transmission of digital image data. Unauthorized access to these image data is simple and easy when they are transmitted via open network. Some of these data contain confidential information, such as military satellite images, patent design blueprints, and medical images. A high security level is required to keep the image data confidential between legal users. The unauthorized accessing should not be able to obtain the content. To protect image security, image encryption approaches are widely studied and applied.

Image encryption techniques attempt to convert the original image into another incomprehensible image. Ciphered image is usually meaningless white noise. Therefore, image encryption is analogous to image scrambling. It requires legal users can easily and efficiently encrypt plain text image and decrypt cipher text image with the knowledge of encryption algorithm and keys. The eavesdroppers without decryption key cannot obtain the content of ciphered images.

JPEG standard is created by Joint Photographic Experts Group. Nowadays, JPEG compression is used in lots of image file formats. Digital image capture devices produce images mostly in JPEG format. It is also the most common format for storing and transmitting images on Internet. To encrypt JPEG images, there are some additional requirements and restrictions. The most significant characteristic of JPEG image compression is the great performance on reducing image file size. Thus, an important principle of JPEG format image encryption is to keep its quality of plain text image and reduced size.

JPEG compression process consists of color space transformation, down sampling, block split, discrete cosine transform (DCT), quantization and entropy coding. After block DCT and quantization, the quantized DC coefficients carry the most important information of every 8*8 block. The process is shown in **Fig. 1**.

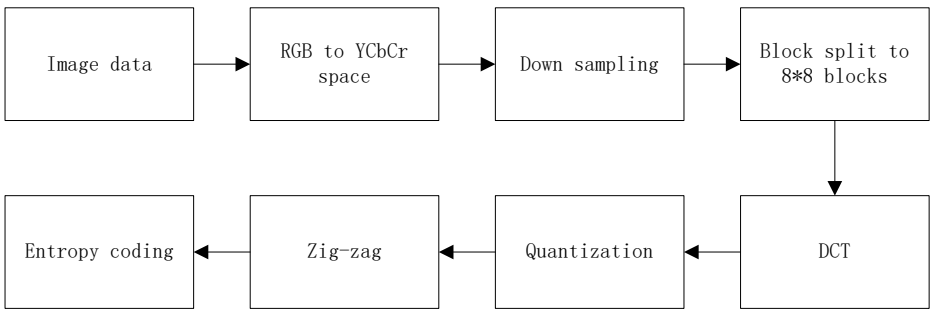


Fig. 1. JPEG encoding process

Based on the Huffman coding principle of JPEG image, this paper proposes a JPEG image encryption scheme that encrypts a quantized DCT coefficient to another value of the same category. Instead of directly encrypting the value of a coefficient, the address of the coefficient is encrypted to get the address of cipher text. Then all 8*8 blocks are shuffled. A chaotic iteration is adopted to generate chaotic sequences for coefficients scrambling and block shuffling. The encryption scheme is resistant to casual observers. Analysis indicates some

shortcomings of the proposed scheme. To overcome these disadvantages, several improvement procedures are further presented in this paper.

2. Related Work

Image encryption has been developed dramatically in recent years. As image data is often compressed by JPEG compression standard, JPEG image encryption has aroused great attention. The encrypted bit stream should keep file format compliance. It means the encrypted data could be decoded as a JPEG image by standard decoder [1].

To keep the image file size, JPEG image encryption algorithms were performed on the DCT domain. Shared key was employed on encrypting the quantized DCT coefficients and the results were also stored in the JPEG format [2]. Some of the existed JPEG image encryption algorithms focused on scrambling DC coefficients. AC coefficients were paid little attention. The DC coefficients were extracted and split to maintain the statistical distribution of DCT coefficients. A spatial temporal-chaos system was utilized to diffuse the quantized DC coefficients [3]. AC signs were encrypted, which made encrypted coefficients not in-relevant enough to the original [4]. Quantized AC positions were shuffled inner blocks, which changed the entropy coding size [5]. To keep the entropy code results inside blocks, all the blocks were shuffled to gain the encrypted image [6]. But the coefficients were not confused. In [7], AC coefficient together with the zero run length was considered as a symbol. The symbol was scrambled. The value of AC was not confused. In [8], Huffman coefficient structure was divided into Huffman code and appended bits. The appended bits were encrypted by XOR operation with bit sequence. DC and AC coefficients were encrypted differently in [9]. DC components were rearranged based on the regions. AC coefficients were ciphered after the smallest bitstream size was selected. Since DC coefficients were usually much larger than AC coefficients in the same 8*8 blocks, Ong et al. decomposed DC coefficients into four parts, placed them in AC positions, and then shuffled these coefficients during all defined blocks [10].

Chaotic system has received increasing attention of image encryption researchers recently because it has many cryptographically desirable features. These features include sensitivity to initial condition, density of periodic orbits, and so on. The chaotic sequences produced by chaotic iteration were employed to shuffle image pixel positions, confuse pixel grey values and scramble frequency coefficients [11][12][13]. Two chaotic maps were used for permutation and substitution processes after discrete Walsh transform [14]. Pseudorandom sequence was adapted to scramble and substitute each 4*4 block [15]. Several JPEG image encryption algorithms also adopted chaotic iteration to generate key sequences [3][6]. In [16], block and symbol scrambling were proposed based on PN sequence that was generated by Pseudo Random Generator.

3. Coefficient Category Address Encryption for JPEG Image

3.1 Chaotic Sequence

Several chaotic sequences could be produced by a multi-dimensional chaotic iteration [17]. Namely

$$(x_{j+1}^1, x_{j+1}^2, ..., x_{j+1}^M) = f(x_j^1, x_j^2, ..., x_j^M; P) \quad (1)$$

in which x_j^k is the *k-th* element in the *j-th* iteration and f is the chosen chaotic iteration with its parameter set P. There will be M chaotic sequences after the iteration. In this paper, two sequences out of these M sequences are chosen for quantized DCT coefficient category address encryption and block shuffle. The key of the whole system consists of the initial value $(x_1^1, x_1^2, ..., x_1^M)$, parameter set P, and the choice of two chaotic sequences.

3.2 Quantized JPEG Coefficient Category

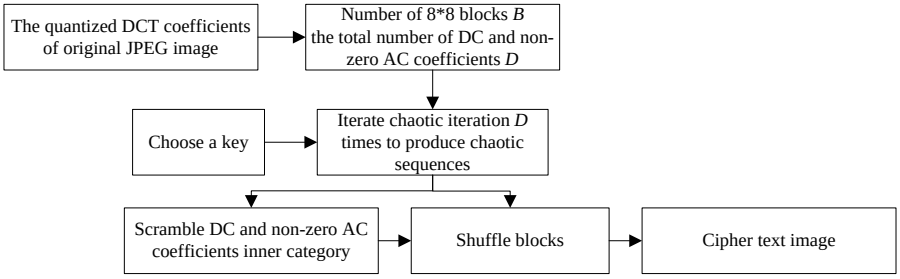
JPEG standard baseline Huffman table codes quantized AC coefficient as two parts: its category together with the zero run length before it and the value. Different values in the same category will be coded in the same length. The processing ensures to express the most common symbols using the shortest code words. Similarly coding process is performed on the difference of quantized DC (DQDC) coefficients from neighbor blocks. **Table 1** is part of the JPEG coefficient coding categories table.

Table 1. Part of JPEG coefficient coding categories

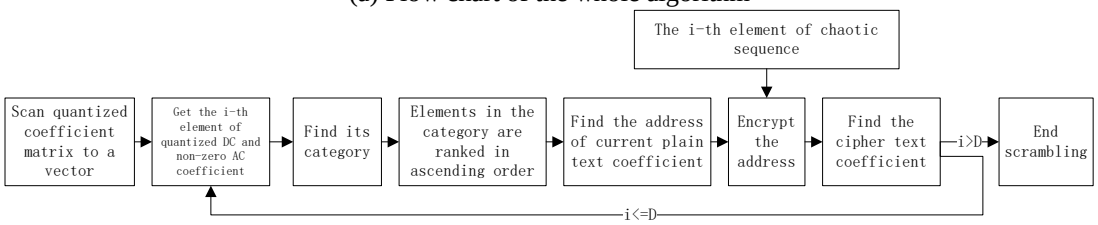
Range	DC difference category	AC category
0	0	N/A
-1,1	1	1
-3,-2,2,3	2	2
……	……	……

3.3 Novel Encryption Scheme

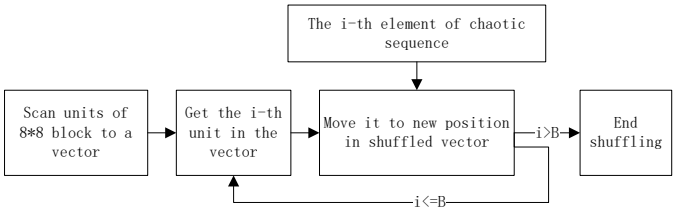
To simplify the procedure of proposed scheme, only luminance coefficients of JPEG images will be encrypted. For colorful images, it is easy to perform the procedure also in color space components. **Fig. 2(a)** is the flow chart of the proposed encryption framework.



(a) Flow chart of the whole algorithm



(b) Flow chart of scrambling process



(c) Flow chart of shuffling process

Fig. 2. Block diagram of the proposed algorithm

The proposed encryption scheme can be described as follows:

1. Choose a key which consists of initial value $(x_1^1, x_1^2, ..., x_1^M)$, parameter set P and the choice of two sequences out of M sequences. The initial value must be in the input range of Eq. (1). The parameter set has to be in a certain range to ensure Eq. (1) is a chaotic iteration. The choice of two from *M* sequences is also part of the key because the permutation and combination enlarge the key space. One-dimensional chaotic iteration could also be adopted to produce only one chaotic sequence. The sequence will be utilized twice. But the reusing of the single sequence makes the key space much smaller. To illuminate this verdict, a simulation and analysis will be presented in section 4.1.
2. Read the quantized coefficients of the original JPEG image, and count the number of 8*8 blocks *B* as well as the total number of DC and non-zero AC coefficients *D*. These two numbers are used to determine the iteration times and length of chaotic sequences in next steps. It is obvious that $D \geq B$, because every block has a DC coefficient.
3. Iterate Eq. (1) for *N* times to get chaotic sequences $\{X^1, X^2, ..., X^M\}$, in which $N = D$, $X^m = (x_1^m, x_2^m, ..., x_N^m)$, $m \in \{1, 2, ..., M\}$. Choose two of them according to the choice in step 1. These two sequences are denoted as $\{X^{sc}, X^{sh}\}$, in which X^{sc} will be used for scrambling and X^{sh} for shuffling.
4. Sort chaotic sequence X^{sc} in ascending order. The elements indices will form a new chaotic sequence Y^{sc} with integer number element. For example, X^{sc} =(0.8760, 0.3910, 0.8573, 0.4405, 0.8873), the ascending order of X^{sc} is (0.3910, 0.4405, 0.8573, 0.8760, 0.8873). Thus, the elements indices are (2, 4, 3, 1, 5), which is the new sequence Y^{sc}.
5. Step 5 to step 9 are scrambling processing. The flow chart of scrambling process is presented in **Fig. 2(b)**. Scan the quantized coefficient matrix of original image column by column.
6. For the *i-th* element of quantized DC and non-zero AC coefficient Q_i, find its category G_i according to **Table 1**. The categories are the same except the first row in **Table 1**. The first row is category zero for quantized DC, in which quantized DC coefficient has the only possible value of zero. Quantized zero AC coefficient does not have any category but the quantized non-zero AC coefficient has a category. In the other categories, quantized DC and non-zero AC coefficients are in the same range. Thus, the quantized DC and non-zero AC coefficients are not discriminated. The elements in G_i are ranked in ascending order. The ascending ordered elements set is C_i. The address of current coefficient Q_i in C_i is A_i, which means $C_i(A_i) = Q_i$. For example, when Q_i =2, the corresponding category G_i =2, and the ascending ordered elements set of category 2 is C_i =(-3, -2, 2, 3). The address of current Q_i in C_i is A_i =3.
7. Encrypt the address A_i by the following calculation

$$E_i = \mathrm{mod}(A_i + Y_i^{sc}, L_{c_i}) \quad (2)$$

in which L_{c_i} is the total number of elements in set C_i and Y_i^{sc} is the *i-th* elements in Y^{sc}. For example of Q_i =2, G_i =2, C_i =(-3,-2,2,3), A_i =3, L_{c_i} = 4. If Y_i^{sc} =3, then $E_i = \mathrm{mod}(A_i + Y_i^{sc}, L_{c_i}) = \mathrm{mod}(3+3,4) = 2$.

8. Find the encrypted coefficient S_i by its index E_i, which means the E_i-th element in set C_i is the encrypted coefficient S_i, $S_i = C_i(E_i)$. Write S_i to the scrambled coefficients matrix in Q_i's original position (r,c). For example, when Q_i =2 and Y_i^{sc} =3, it is easy to find out the encrypted coefficient $S_i = C_i(E_i) = C_i(2)$=-2.
9. Repeat step 5 to 8 until $i > D$. The scrambled coefficients matrix is constructed.
10. Sort the first *B* elements in X^{sh} to get Y^{sh} as in step 4. *B* is the number of 8*8 blocks counted in step 2. The last *D-B* elements in X^{sh} are not used.
11. Step 11 to step 13 are block shuffling processing. The flow chart of block shuffling process is presented in **Fig. 2(c)**. The scrambled coefficients matrix is divided to 8*8 blocks. Every block is considered as a unit. Scan these units column by column. The matrix is transformed to a vector with elements of 8*8 blocks. The vector is marked as *Block*. For example, *Block* = (B_1, B_2, B_3, B_4), in which B_i is an 8*8 block.
12. Shuffle elements of vector *Block* in assistant of Y^{sh} by
$$Sblock(i) = Block(Y_i^{sh}) \quad i = 1,2,...,B \quad (3)$$
in which *Sblock* is the vector after shuffling and Y_i^{sh} is the *i-th* element in Y^{sh}. For example, Y^{sh} =(4, 2, 1, 3), *Block* = (B_1, B_2, B_3, B_4), in which B_i is a 8*8 block. *Sblock* = (B_4, B_2, B_1, B_3). The elements inside any block B_i are not shuffled.
13. Reconstruct a matrix from shuffled 8*8 block vector *Sblock*.
14. Code the encrypted coefficient matrix by Huffman coding with the original Huffman table. Write the coded data into the ciphered JPEG image file. Other parts of the file, such as the image width and height, quantization tables and Huffman tables, are kept the same as the original plain text image file.

Block shuffling is an important processing step of the proposed algorithm. The 8*8 blocks are treated as units and the shuffling is performed between these units. The elements inside any block are not shuffled. There is no requirement to scan elements inside the blocks. The block shuffling in proposed algorithm is different from block scrambling in [15] and [16]. In [15], the sub key was divided into 4*4 blocks to compute quantized DCT coefficient, then each block was scrambled. In [16], quantized DCT coefficients were scanned in zigzag fashion. The position of zigzag scanned DCT blocks were scrambled in accordance with random sequence.

3.4 Decryption Scheme

It is easy to decipher the encrypted image by the proposed scheme. The first five steps are the same as in the encryption processing. Then the scrambled coefficients matrix can be obtained by permuting coefficients matrix blocks inverse to Eq. (3).

For a ciphered coefficient S_i in position (r,c), find its category G_i, the ascending ordered elements set C_i, the total number of elements in the category L_{c_i} and the enciphered address E_i. The original address A_i is calculated by

$$A_i = \begin{cases} E_i - \mathrm{mod}(Y_i^{sc}, L_{C_i}) - 1, & E_i - 1 \geq \mathrm{mod}(Y_i^{sc}, L_{C_i}) \\ E_i + L_{C_i} - \mathrm{mod}(Y_i^{sc}, L_{C_i}) - 1, & else \end{cases} \quad (4)$$

The deciphered coefficient is the A_i -*th* element in set C_i. Write deciphered coefficient to the deciphered coefficients matrix in position (r, c).

3.5 Implement of the Novel Encryption Scheme

In this section, two-dimensional coupled logistic map is applied to generate chaotic sequences. The original JPEG image quantized DCT coefficients are encrypted by the proposed algorithm. The encrypted coefficients are coded and written into the ciphered JPEG image file. The encrypted file is decoded by a standard JPEG decoder to obtain the cipher text image. The encrypted image appears like meaningless noise.

Table 2 lists file size of several encrypted JPEG images and their corresponding original images. The encrypted image size is no greater than 106% of the original one, which is acceptable.

Table 2. Image file size before and after encryption

Image	Original image size	Encrypted image size	File size enlargement
Lena 512*512	31.8k	33.4k	5.0%
Barbara 512*512	42.3k	43.9k	3.8%
Goldhill 256*256	28.0k	28.5k	1.8%
Peppers 512*512	33.3k	35.2k	5.7%
Cameraman 256*256	10.4k	10.9k	4.8%
Couple 256*256	11.5k	11.9k	3.5%
Aerial 512*512	56.0k	58.0k	3.5%
Airfield 512*512	58.7k	60.3k	2.7%
Boats 720*576	49.7k	52.7k	6.0%
Bridge 512*512	61.4k	63.2k	2.9%
Man 1024*1024	154k	161k	4.5%

There are two sources for the enlargement of the size after encryption. The first one is that the proposed scheme encrypts quantized DC coefficients. It destroys the correlation of neighbor quantized DCs and thus makes DQDC bigger in most occasions. For example, the first two of the original quantized DC coefficients are (10, 11). The difference between them is 1. After the coefficients scrambling, these two coefficients might be converted to (-8,15). The difference between them becomes 23, which is much bigger than the original difference.

The second one is that the shuffle will destroy the correlation of quantized DC from neighbor blocks. For example, the first three of the quantized DC coefficients are (10,12,14). The difference between neighbor quantized DC are 2 and 2. If the scrambling does not change the values of these coefficients, shuffling might change the neighborship as (12,10,14). The differences become -2 and 4, which makes the correlation of neighbor DCs varies from original.

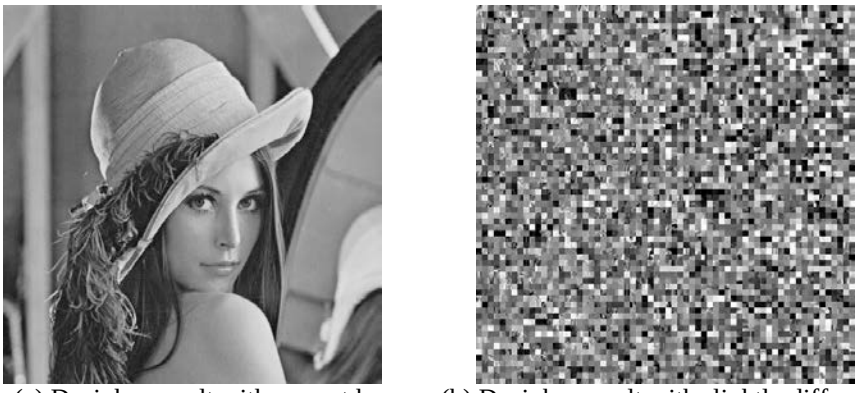
Inner category coefficient encryption could be performed on difference of quantized DC (DQDC) coefficients. However, the offset between ciphered DQDC coefficients and the original accumulates, which might lead to an out ranged quantized DC coefficient. For example, the first two of the original quantized DC coefficients are (DC^{max}, DC^{max} -1), in which DC^{max} is the legally maximum value of quantized DC. The difference between them is

-1. Inner category coefficient encryption might encrypt the difference to 1. This ciphered difference leads to the second quantized DC becoming DC^{max} +1, which is out of range. This phenomenon makes it is impossible for the encrypted data to be decoded to an image by standard JPEG decoder.

3.6 Security Analysis

For casual observers, the only way to attack the encryption is brutal force attack. The key space of proposed scheme is related to the chaotic iteration. For coupled logistic map, the key space is bigger than 100 bits [17]. It is large enough against brute force attack.

If the attacker tries some keys which are very close to the real one, there is still no chance for them to narrow down the possible searching space. **Fig. 3(a)** shows the decryption results of correct and incorrect key. **Fig. 3(b)** presents the decryption result with only 0.000001 difference in one element value of the correct key. It appears still like meaningless noise.



(a) Decipher result with correct key (b) Decipher result with slightly different key

Fig. 3. Decryption result

The attack algorithm presented by [18] counts the number of non-zero quantized DCT elements in every 8*8 block, and then decide to set the whole block to be white or black by the number. The proposed scheme shuffles coefficient matrix blocks to resist this attack. **Fig. 4** provides attack results of encrypted image Lena using proposed scheme. The non-zero coefficients in encrypted image blocks distributed random-like. Therefore, the encrypted image sketch will not be recovered by this attack for the proposed scheme.

Fig. 4. Block non-zero coefficients number attack result of the proposed scheme

If the block shuffle procedure is omitted by the proposed scheme, which means steps 10 to 13 are skiped, the attack will be efficient. **Fig. 5** provides the attack result in this case. It is obvious that sketch of the original image is clear without shuffle.

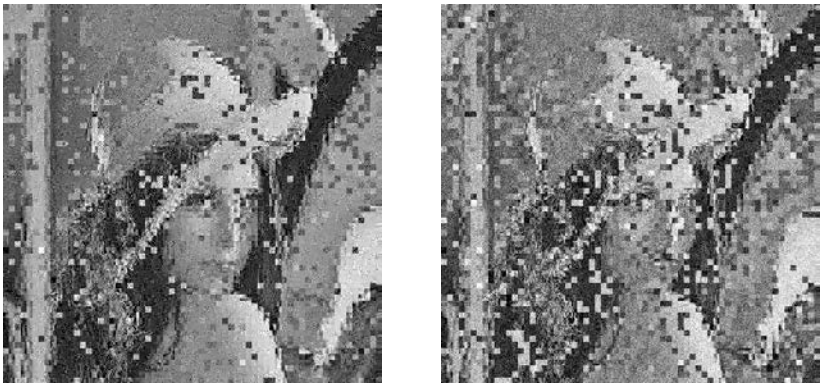
Fig. 5. Block non-zero coefficients number attack result of the proposed scheme without shuffle

3.7 Robustness Analysis

The cipher image will be transmitted via open networks. There might be noises in the networks and the cipher image could be polluted. The most common noises include Gaussian noise, salt and pepper noise. **Fig. 6** presents the decryption results of different noised cipher images. The deciphered images are blurred and not as clear as the original. But the content is still visible.

The explanations of the deciphered image appearance are as following:

1. Noised data will be spread to the whole 8*8 block spatial coefficients. Thus the noises result in block abrupt change of the deciphered images.
2. The DCT quantized coefficients change after the noise affecting cipher text. Some non-zero AC coefficients might be changed to zero, or vice versa. If the number of non-zero quantized AC coefficients are changed, the decryption still process these new coefficients as the same. Thus there will be mistakes on the shift of other coefficients. This will affect the texture apperance of the 8*8 block.



(a) Salt and pepper noise (b) Gaussian noise

Fig. 6. Decryption results of noised cipher images

4. Improvements of the Proposed Scheme

4.1 Utilization of Multi-dimensional Chaotic Iteration

The proposed encryption algorithm employs multi-dimensional chaotic iteration to produce chaotic sequences. The choice of scrambling and shuffling sequences are also part of the key for the whole system. In practical, one-dimensional chaotic iteration could also be adopted to produce chaotic sequence. In this case, there will be only one chaotic sequence. The proposed algorithm requires two sequences for scrambling and shuffling, respectively. Thus, the only sequence will be utilized twice.

Fig. 7 shows the encryption result of JPEG image Lena with one-dimensional chaotic iteration. The chaotic iteration is 1-d logistic map. The encrypted image size is 33.4 k. The appearance of cipher text is also random like. The possible space of initial value of logistic map and coupled logistic map are provided in **Table 3**. It is obvious that two-dimensional chaotic iteration has much bigger possible space for initial value. The greater possible space indicates more resistance to brutal force attack.

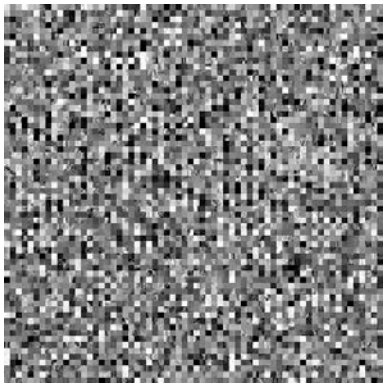


Fig. 7. Encryption result with one-dimensional chaotic iteration

Table 3. Possible space for initial value

Logistic map	Coupled logistic map
2^{52}	2^{104}

Furthermore, the proposed algorithm recommends to use multi-dimensional chaotic iteration because the choice of sequences will also enlarge the key space. For an M-dimensional chaotic iteration, there will be M^2 possible combinations of selected two sequences. For example, coupled logistic map will produce two chaotic sequences $\{X^1, X^2\}$ after the iteration. There will be four combinations of selected sequences for scrambling and shuffling: $(X^1, X^1), (X^1, X^2), (X^2, X^1), (X^2, X^2)$. But for 1-d logistic map, there will be only one possible combination: (X^1, X^1). Thus, the utilization of multi-dimensional chaotic iteration avails a greater key space. It has greater possible initial value space, and enlarges the key space by the combination of chaotic sequences for scrambling and shuffling.

4.2 Category Merging

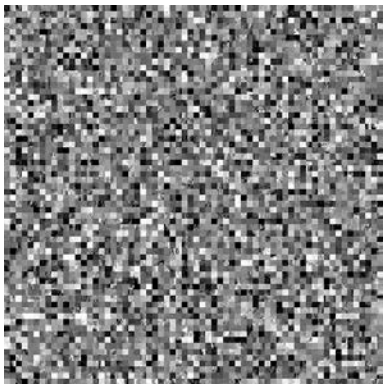
The proposed encryption algorithm shifts quantized DC and non-zero AC coefficients inside categories. The first several categories include small numbers of elements, which means the encrypted element suffers from small possible space. A solution to this problem is to merge neighbor categories to one class.

The merged class has more elements than original categories, which enlarges the cipher text possible space. For example, if the observer obtains a cipher text -1, he/she will know that the plain text has only two possible values: -1 and 1. But, after merging category 1 and 2 to be one class, the observation of cipher text -1 means the plain text has six possible values. The category merging procedure should be added between step 5 and step 6:

Merge neighbor categories to one class. The merging combination choice constitutes part of the key.

After the merging procedure, G_i indicates the class instead of the category of Q_i. **Fig. 8** presents the encryption results of image Lena by the proposed scheme with category merging procedure. In this simulation, only category 1 and category 2 are merged into a class.

As the block shuffling processing is kept as the original scheme, encryption with category merging is also resistant to the attack algorithm presented by [18]. **Fig. 8(a)** shows the encryption results of image Lena by the proposed scheme with category merging procedure. **Fig. 8(b)** provides block non-zero coefficients number attack result of **Fig. 8(a)**. The encrypted image sketch is not recovered by this attack. If the block shuffle procedure is omitted, the attack will be efficient. **Fig. 8(c)** provides the attack result in this case. It is obvious that sketch of the original image is clear without block shuffle.



(a) Encryption result

(b) Attack result of (a)

(c) Attack result on category merging process without shuffle

Fig. 8. Encryption and block non-zero coefficients number attack results with category merging

Table 4 provides the file size of several JPEG images before and after encryption with category merging. Compared with the last columns in **Table 2**, it is clear that merging makes the encrypted image file size greater. The reason of file size increase is that merged class has more possible values for cipher text. It enlarges the DQDC of ciphered image. Besides, it also

causes the AC values change more. The file size enlargement of proposed scheme with category merging procedure is less than 20%.

Table 4. Image file size of encryption with category merging

Image	Original image size	Encrypted image size	File size enlargement
Lena 512*512	31.8k	37.2k	17.0%
Barbara 512*512	42.3k	48.3k	14.2%
Goldhill 256*256	28.0k	30.5k	8.9%
Peppers 512*512	33.3k	39.5k	18.6%
Cameraman 256*256	10.4k	12.1k	16.3%
Couple 256*256	11.5k	13.0k	13.0%
Aerial 512*512	56.0k	63.1k	12.7%
Airfield 512*512	58.7k	66.9k	14.0%
Boats 720*576	49.7k	57.8k	16.3%
Bridge 512*512	61.4k	70.0k	14.0%
Man 1024*1024	154k	179k	16.2%

Since the category merging procedure leads to more elements in one class, the coefficient address searching needs more efforts. This results in more excution time. The execution time of these schemes is given in **Table 5**. The computer used for this simulation is with Intel(R) Core(TM) i5-2430M CPU@2.40G Hz. The scheme with category merging needs more time to perform encryption for the same image. The file size enlargement and excution time enlongation are the price for enlarged cipher text possible space.

Table 5. Execution time with and without category merging

Image	Without category merging	With category merging
Lena 512*512	0.932715s	1.280580s
Barbara 512*512	1.241642s	1.721441s
Goldhill 256*256	0. 718670s	0.964972s
Peppers 512*512	0. 995488s	1.358826s
Cameraman 256*256	0.294464s	0.414961s
Couple 256*256	0.348369s	0.418039s
Aerial 512*512	1.667096s	2.195181s
Airfield 512*512	1.775586s	2.293438s
Boats 720*576	1.415756s	2.058354s
Bridge 512*512	1.879488s	2.464360s
Man 1024*1024	4.698607s	6.224267s

4.3 Unidirectional and Bidirectional Diffusion

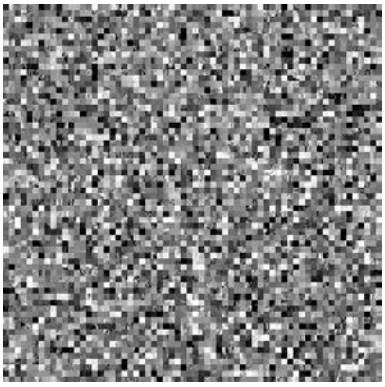
There are two main procedures in the proposed encryption algorithm: scrambling and shuffle. The proposed scheme has no diffusion processing. If a quantized DC or non-zero AC coefficient in the JPEG image changes, the change will not spread. This makes the proposed algorithm vulnerable to the differential attack. The attackers know the encryption procedure and could obtain cipher image corresponding to designated input image. Thus they may change one coefficient at a time to see the difference of cipher image. The pattern of confusion and shuffle for every coefficient will be revealed. Then images encrypted by the same key will be recovered easily. The best way to resist differential attack is to spread slight change of plain

text to other cipher texts.

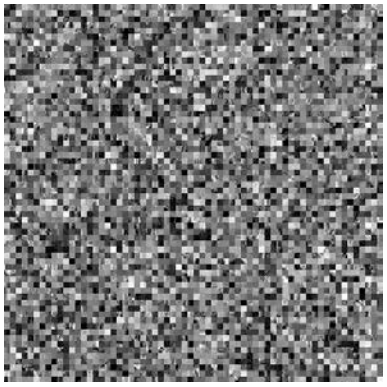
To spread the change of plain text in cipher text, diffusion is introduced into the encryption processing. The current cipher text is produced by current plain text, cipher key and former plain or cipher text. This unidirectional diffusion procedure could be implemented along with the confusion, which means Eq. (2) in step 7 is changed to

$$E_i = \mathrm{mod}(A_i + Y_i^{sc} + E_{i-1}, L_{c_i}), \quad i = 1, 2, ..., N \tag{5}$$

For the first plain text encryption, E_0 is also a part of the key, which could be set to any integer. The procedure enlarges the key space. **Fig. 9(a)** presents the encryption result of JPEG image Lena by proposed scheme with unidirectional diffusion.



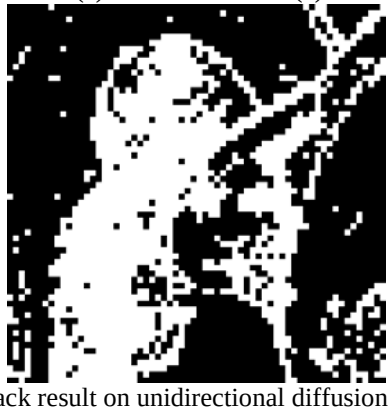
(a) Encryption result with unidirectional diffusion



(b) Encryption result with bidirectional diffusion.

(c) Attack results of (a)

(d) Attack results of (b)



(e)Attack result on unidirectional diffusion process without block shuffle

(f) Attack result on bidirectional diffusion process without block shuffle

Fig. 9. Encryption and block non-zero coefficients number attack results with diffusion

There is an inherent shortcoming of the unidirectional diffusion procedure. The spread effect is related to the position of changed plain text. If the former plain text changes, the change will spread to all the cipher text after the current one. If the last plain text changes, the other cipher text will not change at all. To overcome this shortcoming, bidirectional diffusion could be adopted [19]. After the confusion processing, bidirectional diffusion will be applied. Namely the step **7** in section 3.3 is replaced by:

Diffuse the encrypted coefficient index bidirectional, which means two diffusion processing are performed sequentially. In other words,

$$\begin{cases} E_i^0 = \mathrm{mod}(A_i + Y_i^{sc} + E_{i-1}^0, L_{c_i}), & i = 1,2,...,N \\ E_i^1 = \mathrm{mod}(E_i^0 + E_{i+1}^0, L_{c_i}) \quad , & i = N,...,2,1 \end{cases} \tag{6}$$

in which E_i^0 is the *i-th* encrypted element after orderly encryption with diffusion, E_i^1 is the *i-th* encrypted element after diffusion inverse orderly. E_0^0 and E_{N+1}^0 are also a part of the key, which could be set to any integer.

After the bidirectional diffusion, any changed plain text will affect all of the cipher text. This makes it computationally expensive for differential attack. **Fig. 9(b)** shows the encryption result of JPEG image Lena by the proposed scheme with bidirectional diffusion.

The block shuffling processing is kept as the original scheme for both diffusion procedures. Thus, both of the improved encryption schemes are resistant to the attack algorithm presented by [18]. **Fig. 9(c)** and **Fig. 9(d)** provide block non-zero coefficients number attack results of **Fig. 9(a)** and **Fig. 9(b)**. The encrypted image sketchs are not recovered. If the block shuffle procedure is omitted, the attack will be efficient. **Fig. 9(e)** provides the attack result on unilateral diffusion process without block shuffle. **Fig. 9(f)** shows the attack results on bilateral diffusion process without block shuffle. The sketch of the original images are clear in this case. It is obvious that block shuffle procedure is very important for the security of the algorithm.

As the bidirectional diffusion processes all coefficients twice, it needs more time to execute. The execution time of these schemes is given in **Table 6**. It is clear that bidirectional diffusion needs the most time. That is the price for higher level of security.

Table 6. Execution time of difusion

Image	Unilateral diffusion	Bilateral diffusion
Lena512*512	0.980073s	1.593730s
Barbara 512*512	1.267377s	2.105883s
Goldhill 256*256	0. 752545s	1. 193198s
Peppers 512*512	1. 019195s	1. 618143s
Cameraman 256*256	0.367418s	0.563723s
Couple 256*256	0.426337s	0.612730s
Aerial 512*512	2.009801s	3.070198s
Airfield 512*512	2.147700s	3.373260s
Boats 720*576	1.757703s	2.718394s
Bridge 512*512	2.256288s	3.526085s
Man 1024*1024	5.805816s	8.937039s

The file size of images after encryption with diffusion procedures are provided in **Table 7**. The cipher image size in the second and fourth columns are little different from the third column of **Table 2**. And the cipher image size in the third and fifth columns are almost the same as the third column in **Table 4**. It is because that diffusion procedure does not affect the

address searching space. The psudo-randomness of chaotic sequence makes the cipher text distribute uniformly in possible space. Thus the cipher image size only depends on the cipher text possible space, not on diffusion procedure.

Table 7. Cipher image file size with diffusion

Image	Unilateral diffusion		Bilateral diffusion	
	Without catogery merging	With catogery merging	Without catogery merging	With catogery merging
Lena512*512	33.3k	36.9k	33.4k	36.9k
Barbara 512*512	43.9k	48.3k	43.8k	49.9k
Goldhill 256*256	28.5k	30.5k	28.6k	30.5k
Peppers 512*512	35.2k	39.5k	35.2k	39.4k
Cameraman 256*256	10.9k	12.0k	10.9k	12.0k
Couple 256*256	11.9k	13.1k	11.9k	13.0k
Aerial 512*512	58.0k	63.2k	58.0k	63.3k
Airfield 512*512	60.3k	66.9k	60.3k	67.4k
Boats 720*576	52.7k	57.7k	52.7k	58.0k
Bridge 512*512	63.2k	69.9k	63.2k	70.0k
Man 1024*1024	161k	179k	161k	179k

5. Conclusion

This paper proposes a JPEG image encryption scheme that encrypts quantized DC coefficients as well as non-zero quantized AC coefficients. The coefficient value is transferred into another one in the range of the same category. The key to the whole system includes the initial value, the parameter set of a chaotic iteration and choice of chaotic sequences. Analysis of experimental results validates the efficiency and effectiveness of the proposed scheme. To overcome the shortcomings of the proposed scheme, category merging and diffusion procedures are embedded into the scheme. These actions could enlarge the cipher text possible space. The improved schemes are more resistant to sophisticated attacks with acceptable cost.

References

[1]. Jolly Shah and Vikas Saxena, "Performance Study on Image Encryption Schemes," *IJCSI International Journal of Computer Science Issues*, Vol. 8, Issue 4, No 1, pp.349-355, July 2011. Artical (CrossRef Link).

[2]. Subramania Sudharsanan, "Shared Key Encryption of JPEG Color Images," *IEEE Transactions on Consumer Electronics*, Vol. 51, No. 4, pp.1204-1211, November, 2005. Artical (CrossRef Link).

[3]. Yuling Luo, Minghui Du, Dong Liu, "JPEG Image Encryption Algorithm Based on Spatiotemporal Chaos," in *Proc. of 2012 Fifth International Workshop on Chaos-fractals Theories and Applications*, pp. 191-195. 2012. Article (CrossRef Link).

[4]. Qiu Jing and Wang Ping, "Encryption Algorithm for Compressed Image Based on Chaotic Maps," *Computer Science*, Vol. 39, No. 6, pp.44-46, June, 2012. Artical (CrossRef Link).

[5]. Tang, L., "Methods for encrypting and decrypting MPEG video data efficiently," in *Proc. of the Fourth ACM International Conference on Multimedia*, pp. 219-229. 1996. Article (CrossRef Link).

[6]. Zhang, Dinghui, and Fengdeng Zhang, "Chaotic encryption and decryption of JPEG image," *Optik-International Journal for Light and Electron Optics* 125.2, pp.717-720, 2014. Article (CrossRef Link).
[7]. B. K. ShreyamshaKumar and Chidamber R. Patil, "JPEG Image Encryption using Fuzzy PN Sequences," *Signal, Image and Video Processing*, Vol. 4, No. 4, pp.419-427, November, 2010. Article (CrossRef Link).
[8]. Zhengxing Qiang, Xinpeng Zhang, Shuozhong Wang, "Reversible Data Hiding in Encrypted JPEG Bitstream," *IEEE Transactions on Multimedia*, Vol.16, No.5, pp.1486 – 1491, August, 2014. Article (CrossRef Link).
[9]. SimYing Ong, KokSheik Wong, Xiaojun Qi, Kiyoshi Tanaka, "Beyond format-compliant encryption for JPEG image," *Signal Processing: Image Communication*, Volume 31, February, Pages 47-60, 2015. Artical (CrossRef Link).
[10]. Fangchao Wang and Sen Bai, "JPEG Image Encryption by Shuffling DCT Coefficients in Defined Block," in *Proc. of Computational and Information Sciences (ICCIS), 2013 Fifth International Conference on* , pp.60 – 63, 2013. Article (CrossRef Link).
[11]. Lian S, Sun J,Wang Z, "A block cipher based on a suitable use of the chaotic standard map," *Chaos, Solitons &Fractals*, 26, 1, pp.117–129, 2005. Article (CrossRef Link)
[12]. Yang L, Yang-yu F, Chong-yang H, "Information hiding technology based on image second scrambling," *Journal of Image and Graphics*, Vol.11, No. 8, pp.1088–1091, 2006. Artical (CrossRef Link).
[13]. Zeng W, Lei S, "Efficient frequency domain selective scrambling of digital video," *IEEE Trans Multimedia*, Vol.5, No.1, pp.118–129, 2003. Article (CrossRef Link).
[14]. Telem, Adelaide Nicole Kengnou, et al., "A Robust Chaotic and Fast Walsh Transform Encryption for Gray Scale Biomedical Image Transmission," *Signal and Image processing: an international journal*, volume 6(Issue 3):81-102, 2015. Article (CrossRef Link).
[15]. Padmapriya Praveenkumar, Rengarajan Amirtharajan, K. Thenmozhi and John Bosco Balaguru Rayappan, "Pixel scattering matrix formalism for image encryption—A key scheduled substitution and diffusion approach," *AEU-International Journal of Electronics and Communications*, Volume 69, Issue 2, February, pp. 562-572, 2015. Article (CrossRef Link).
[16]. Kishore, B., Shreyamsha Kumar, B. K., Patil, C.R. "FPGA based simple and fast JPEG Encryptor," *Journal of Real-Time Image Processing*, 10 (3), pp. 551-559. 2015. Article (CrossRef Link).
[17]. Shanshan Li, Yinghai Zhao, Bayi Qu and Jiang-an Wang "Image scrambling based on chaotic sequences and Veginère cipher," *Multimedia Tools and Application*. Vol.66, No.3, pp. 573-588, 2013. Article (CrossRef Link).
[18]. WEIHAI LI, YUAN YUAN, "A leak and its remedy in JPEG image encryption," *International Journal of Computer Mathematics*, Vol. 84, No.9, pp. 1367–1378, September, 2007. Article (CrossRef Link).
[19]. Chong Fu, Jun-jie Chen, Hao Zou, Wei-hong Meng, Yong-feng Zhan and Ya-wen Yu, "A chaos-based digital image encryption scheme with an improved diffusion strategy," *Opetics Express*, Vol.20, No.3, pp. 2363-2378, January, 2012. Article (CrossRef Link).

Shanshan Li received her Ph.D. in signal and information processing, University of Science and Technology of China. Her research interest includes image processing, image security, image analysis, etc. Now she is an associate professor in Department of Electronic Information Engineering, School of Information Engineering, Chang'an University.

Yuanyuan Zhang is an M.D. candidate in clinic medicine, Xi'an Jiaotong University. Her research interest includes image assisted tumor diagnosis.