A novel Image Security Protection Method Based on DCT Compression Theory and Hyper-chaotic Mapping

Xianglian Xue1,2, Haiyan Jin1,3,*

1 School of Computer Science and Engineering, Xi’an University of Technology, Xi’an, 710077, Shaanxi, China.
2 Shaanxi University of Chinese Medicine, Xianyang, 712046, Shaanxi, China.
3 Shaanxi Key Laboratory for Network Computing and Security Technology, Xi’an University of Technology, Xi’an, 710077, Shaanxi, China.

*Corresponding author e-mail: jinhaiyan@xaut.edu.cn

Abstract. This paper studies the current situation of image compression encryption and analyzes the influence of low frequency (DC coefficient) and high frequency (AC coefficient) on image structure in DCT transformation. Based on this, a novel image security protection method based on DCT compression theory and hyper-chaotic mapping is proposed. First, the position of the pixel of the original image is disturbed, and converts the image from spatial domain into frequency domain by the two-dimensional DCT transformation and quantization. Second, change the pixel values by modifying the values of the sign bit of AC coefficient and DC coefficient. At last, the encrypted image is obtained by carrying out inverse quantization, inverse transformation and reverse operation by bit.

Key words: Image Encryption; DC Coefficient Encryption; AC Coefficient Encryption; Security

1. Introduction

Nowadays, with the limitation of network bandwidth and the increasing resolution of image acquisition equipment, most of the images in the network are transmitted in compressed format. For some confidential images, the security of the image itself must be considered while pursuing the transmission efficiency, so the algorithm mechanism of "compression and encryption simultaneously" is undoubtedly one of the main means to solve this kind of problem.

Since the JPEG (Joint Photographic Experts Group) coding format has the advantages of large compression ratio and high compression quality [1], it is widely used in image compression. As a result, image encryption algorithm based on JPEG compression mechanism has become a hot topic. Wu[2] encrypt the compressed image by changing the symbol bits of the DC and AC coefficients in the DCT transform domain. This method only changes the symbol bits of the coefficients in each image block, and does not change the value of the correlation coefficients. Lian[3] et al analyzed the Wu method and obtained the original image. Guo [4] proposed an image encryption algorithm combined with JPEG compression. For the DC and AC coefficients in the domain, the encrypted
compressed image was obtained by grouping and scrambling with Logistic chaotic mapping. However, the low-dimensional Logistic chaotic mapping has a simple structure, a small number of initial parameters and smaller range of key values, which leads to a low security of encryption [5-7]. Lu [8] et al. obtained the encrypted images by scrambling DC and other coefficients, and achieved good encryption effect, but the algorithm did not consider the expansion of run-length coding, resulting in a decrease in compression ratio. Han [9] proposed a JPEG image encryption algorithm based on DC coefficient hiding, which hides the DC coefficient after deformation into the scrambled AC coefficient, realized the image compression encryption. The algorithm has little effect on the compression ratio, but the encryption effect is not ideal. An image security protection method based on DCT compression theory and hyper-chaotic system is proposed in this paper, which can achieve higher security and better hiding effect of original image information and of course, this algorithm can be used as the interface of encryption and compression joint algorithm.

2. Hyper-chaotic mapping

Compared with the combination of low-dimensional chaotic mapping or low-dimensional chaotic mapping, it is an indisputable fact that hyper-chaotic mapping has absolute superiority. Xue et al [10] think that the fractional-order Chen Hyper-chaotic (FOCHC) mapping based on Adomain is one of the best systems. We use FOCHC mapping to complete the image information encryption. The model of the FOCHC system is defined as follows:

\[
\begin{align*}
\frac{d^q x}{dt^q} &= m(y - x) + w \quad \frac{d^q y}{dt^q} = nx - xz + oy \\
\frac{d^q z}{dt^q} &= xy - pz \\
\frac{d^q w}{dt^q} &= yz - sw
\end{align*}
\]  

Set \(m = 38, n = 7, o = 12, p = 3, s = 0.7, \) and \( q = 0.95, \) Chen system goes into chaos, and generates four pseudorandom sequences \(x, y, z, w\) simultaneously.

3. Algorithm analysis and description

The proposed algorithm in this paper is divided into three parts, first disrupting the position of image pixels. Then it is DCT transformed and quantized, the extracted DC coefficients are XOR and the extracted AC coefficients are modified symbol bit. Finally, the inverse DCT transformation and the inverse quantization matrix are taken to diffuse the pixel value of the image. Figure 1 shows the structured flowchart of the presented method. Some analysis of the proposed algorithm: (1). When the image is quantized, we should select the appropriate quantization factor, the larger the quantization factor, the greater the compression ratio, the image will lose a lot of data in the process of encryption, resulting in the image distortion obtained during decryption. The algorithm encrypts the bitmap image without emphasizing the compression ratio of the encrypted image, so the quantization factor is 0.1, the image is almost uncompressed in the process of encryption, and the decryption image is the same as the original image. Of course, this algorithm can be used as the interface of encryption and compression joint algorithm. (2). DC coefficient describes the main information of the image, and the deformed chaotic sequence and the DC coefficient are implemented XOR operation to modify the DC coefficient. (3). The AC coefficient describes some edges and contours of the image, but the AC coefficient has a fixed range of values, if beyond the range, the result of inverse quantization and inverse transformation is not ideal. Therefore, we use a pseudorandom chaotic sequence to modify the symbol bit of each AC coefficient. (4). The DC, AC coefficient will appear block effect after DCT transformation. In this paper, reverse operation is used to eliminate block effect.
Figure 1. Structured flowchart of the presented method

Combined with the Figure 1, the method is executed according to the following steps:

Step 1: Enter a gray image \( I(m,n) \);

Step 2: Four chaotic sequences \( x, y, z, w \) are generated using fractional Chen hyper-chaotic mapping under initial value \( 0, y_0, z_0, w_0 \);

Step 3: A sort function \( \{l_x, f_x\} = \text{sort}(\cdot) \) is used to sort \( x, y \) sequence obtained from step 2, and the index value of the sorting result is taken. The sort(\( ) \) can sort a sequence from small to large and return the new sequence and index value after sorting. The return value of the new sequence is represented by \( f_x \) and the index value of corresponding to the new sequence is represented by \( l_x \).

Step 4: \( I(i,j) \) is replaced by \( I(l(x(i),y(j))) \) to scramble \( I(m,n) \), after scrambling, get the matrix \( I_s(m,n) \);

Step 5: Divided \( I_s(m,n) \) into \( 8 \times 8 \) block, then, Executed DCT transformation, and quantified by the following formula:

\[
L = Q \times \begin{bmatrix}
16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\
12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\
14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\
14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\
24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\
49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\
72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \\
34 & 10 & 12 & 14 & 16 & 24 & 40 & 51
\end{bmatrix} \tag{2}
\]

Where the quantization factor \( Q = 0.1 \), and the quantization matrix is \( I_l(i,j,k) \), where \( i = 1,2 \cdots 8, j = 1,2 \cdots 8, k = m \times n / 8 \times 8 \).

Step 6: The first element of each block in the matrix \( I_l(i,j,k) \) is extracted according to formula (3) as the DC coefficient:

\[
DC(k) = I_l(1,1,k) \quad k = 1,2 \cdots m \times n / 8 \times 8 \tag{3}
\]

Step 7: According to formula (4), the DCZ sequence is obtained. Where \( x \) is chaotic sequence which got from step 2, \( \text{mean}(\cdot) \) is a mean function, \( \text{mod}(\cdot) \) is a function of finding a remainder, \( \text{abs} \) is an absolute function, \( \text{bitxor}(\cdot) \) is a XOR function by bit.

\[
\begin{align*}
\{x' = \text{mod}(x \times 10^4, \text{mean}(DC)) \\
\text{DCZ} = \text{bitxor}(\text{abs}(DC), x')
\end{align*} \tag{4}
\]

Step 8: Deform the \( y \) sequence into 0,1 sequence \( y' \), modify the symbol bit according to formula (5) for the AC coefficient in each block of matrix \( I_l(i,j,k) \).

\[
\begin{align*}
\{AC(i,j) = I_l(i,j,k) \ast (-1)^{y(j)} & = 1,2 \cdots 8, j = 1,2 \cdots 8, k = 1,2 \cdots m \times n / 8 \times 8, i = 1,2 \cdots 8, i = 1,2 \cdots 8, k = 1,2 \cdots m \times n / 8 \times 8, i = 1,2 \cdots 8, i = 1,2 \cdots 8, k = 1,2 \cdots m \times n / 8 \times 8
\end{align*} \tag{5}
\]

Step 9: The DC coefficient matrix and AC coefficient matrix block after encryption are
reconstructed, and the matrix $I_g$ is obtained by inverse quantization and inverse DCT transformation.

Step10: The following algorithm is used to normalize the $I_g$ to locate the pixel value at 0~255, the cipher-text $P$ is obtained.

```plaintext
if $I_g(m,n) \geq 0$ and $I_g(m,n) < 255$
    $I_g(m,n) = I_g(m,n)$;
    $p(m,n) = 0$;
elseif $I_g(m,n) < 0$
    $I_g(m,n) = abs(I_g(m,n))$;
    $p(m,n) = -1$
else
    $I_g(m,n) = mod(I_g(m,n),256)$;
    $p(m,n) = floor(I_g(m,n)/256)$;
endif
endif
return $p$
```

Step11: The chaotic sequence $z$ is transformed into a matrix of size $(m \times 4,1)$, sequence $w$ is transformed into a matrix of size $(1,n \times 2)$, $z$ and $w$ multiply to obtain matrix $Z$ of size $(m \times 4, n \times 2)$, deformed $Z$ into one-dimensional matrix $e$. Then each element after the normalized matrix is converted to an 8-bit binary number, the sequence $e$ and normalized matrix $I_g'$ are reversed according to the following algorithm. After regroup, obtained the encrypted image $I'$.

```plaintext
if $e(\tau) < 0.5$
    if $I_g'(\tau) == '1'$
        $I_g'(\tau) = '0'$
    elseif $I_g'(\tau) == '0'$
        $I_g'(\tau) = '1'$
endif
endif
```

During image decryption, the receiver obtains the cipher-text $P$ and the correct key from the encryption side at first and the corresponding decryption image can be obtained according to the inverse operation of the image encryption algorithm.

4. Analysis of simulation results
To illustrate the effectiveness of the proposed method, set $x_0 = 0.31, y_0 = 0.42, z_0 = 0.53, w_0 = 0.67$, three grayscale images lenna, baboon and pepper size of $512 \times 512$ (see Figure 2 (a)(d)(g)) are encrypted. The corresponding encryption result see Figure2 (b)(e)(h), and Figure2 (c)(f)(i) shows the decrypted image. Obviously, the encrypted image (see Figure2 (b) (e) (h)) is completely different from the corresponding original image (see Figure2 (a)(d)(g)), however, the decrypted image (see Figure2 (c)(f)(i)) is no different from the original image.
5. Security analysis

5.1. Key-space analysis

The four initial values $x_0, y_0, z_0, w_0$ of FOCHC taken as keys, set computer precision as $10^{-14}$, the key space is $10^{14} \times 10^{14} \times 10^{14} \times 10^{14} = 10^{64}$. Therefore, the key space of this algorithm is $10^{14} \times 10^{14} \times 10^{14} = 10^{42} \approx 2^{202}$. In addition, cipher-text $P$ is also unknown. To sum up, the key space of this algorithm can resist the attack of exhaustive method.

5.2. Key sensitivity analysis

Because the sensitivity of the initial condition is one of the main distinguishing features of the FOCHC, which can lead to there is a large difference between decrypted images when a small change in decryption key. Similarly, there is a large difference between encrypted images when a small change in encryption key. The test conclusion is shown in Figure 3. The decryption key is changed to $x_0 = 0.3000000001$, while the other keys are unchanged. The decryption algorithm is used to decrypt Figure 2 (b). As shown in Figure 3 (a), the resulting image is completely different from the original image. Minor changes to other keys do not restore the original image, which is no longer repeated here. Figure 3 (b) is the decryption image of the wrong cipher-text $P$. Obviously, the encryption key is sensitive, so it is hard to analyze the plaintext image by exhaustive analysis.
Figure 3. The test results of key sensitivity. (a) The decryption image when \( x_0 = 0.300000000000001 \) with other keys unchanged; (b) The decrypted image under wrong cipher-text P with other keys unchanged.

5.3. Histogram Analysis

It is one of the main methods to get meaningful information by using calculate distribution of the pixel value from encrypted image. To statistics the pixel distribution, we give the gray histograms of lenna, baboon and peppers images and the gray histograms of the corresponding encrypted images, shown as Figure4. It can be compared that the pixels of the images are concentrated around the 50,100,150 before the image encryption, and the gray histogram values after encrypting are close to 100, and the distribution is very uniform. It is proof that the similarity between the original image and the image after encrypted is very low, and it is hard to get the original image through statistical analysis.

Figure 4. Histogram Analysis. (a) ‘lenna’ gray histogram; (b) The gray histogram after encrypting for ‘lenna’; (c) ‘baboon’ gray histogram; (d) The gray histogram after encrypting for ‘baboon’; (e) ‘peppers’ gray histogram; (f) The gray histogram after encrypting for ‘peppers’.

5.4. Analysis of adjacent pixel correlation

For the calculation of correlation coefficients, see formulas (6)-(8)

\[
W(m) = \frac{1}{M} \sum_{i=1}^{M} (m_i - \frac{1}{M} \sum_{i=1}^{M} m_i)^2
\]

(6)

\[
C(m, n) = \frac{1}{M} \sum_{i=1}^{M} \left( m_i - \frac{1}{M} \sum_{i=1}^{M} m_i \right) \left( n_i - \frac{1}{M} \sum_{i=1}^{M} n_i \right)
\]

(7)

\[
R_{mn} = \frac{C(m, n)}{\sqrt{W(m) \cdot W(n)}}
\]

(8)
The function $C(m,n)$, $W(m)$, $R_{mn}$ are the covariance, variance and correlation coefficient, respectively. The parameter $m$ and $n$ is the value of two adjacent pixels in the horizontal direction (or vertical, or diagonal) of the image. The range of $(m,n)$ is the random extraction of 3000 pairs of pixels in different directions.

**Table 1.** Correlation coefficient of the original and encrypted image of ‘lenna’ for three directions

|         | Original image | Encrypted image |
|---------|----------------|-----------------|
| Horizontal | 0.9721         | 0.0014          |
| Vertical  | 0.9859         | 0.0074          |
| Diagonal  | 0.9555         | 0.0088          |

From Table 1, we can see the correlation coefficients of each two adjacent pixels of the original image of ‘lenna’ in three different directions are 0.9721, 0.9859, 0.9555, respectively, while the correlation coefficients of the encrypted image of ‘lenna’ are 0.0014, 0.0074 and 0.0088, respectively. It can be seen from the data that the correlation between each two adjacent pixels in different directions is very low. Similarly, the 3000 pairs in Figure 5(a) are distributed around the diagonal direction, and the 3000 pairs in Figure 5(b) are scattered in all directions, and the distribution is very messy. The strong ability of anti-statistical analysis can be obtained from the data in Table 1 and the scattered point analysis in Figure 5.

![Figure 5. Correlation between two adjacent pixels in the horizontal direction for the original image and the encrypted image: (a) Correlation between two adjacent pixels in the horizontal direction of the original image lenna. (b) Correlation between two adjacent pixels in the horizontal direction of the encrypted image lenna.](image)

5.5. Information entropy analysis

By using the information entropy (IE) theory in the Ref.[10], to verify encryption effect. The IE of the encrypted images is shown in Table 2, and all values of IE are very close to 8.

**Table 2.** The IE of encrypted Image

| The encrypted image | Information entropy |
|---------------------|---------------------|
| Lena                | 7.9991              |
| Baboon              | 7.9988              |
| Peppers             | 7.9984              |

5.6. Against noise attack analysis

Add 10% salt and pepper noise to the encrypted image which shown in Figure 6 (a), Figure 6 (b) shows the decrypted effect of Figure 6 (a). And the signal-to-noise ratio is 20.0334, we can see the details and...
basic contours of the original image can be clearly identified from Figure 6 (b).

**Figure 6.** Test of against salt and pepper noise attack. (a) Add 10% salt and pepper noise to encrypted image. (b) The decrypted image of (a).

### 6. Conclusions

A new based on DCT compression theory and hyper-chaotic mapping image security protection method is proposed. We evaluate the effect and security of the method from three aspects: first, the IE of all the encrypted images are ideal value and the dense encrypted images that can not see any information can be obtained. Second, the space estimation and sensitivity test of the keys can proof that the method can resist the exhaustive attack effectively. Third, from the "evenly distributed gray histogram" and "very close to 0 correlation coefficient" (see Section 4.3 and Section 4.4), the statistical analysis is also helpless to the proposed algorithm in this paper. In addition, 10% salt and pepper noise attack can still obtain a more ideal decryption image.

### Acknowledgments

Shaanxi Province Technology Innovation guidance Special Project (NO:2020CGXNG-026).

### References

[1] Chen J.Z., Bao Y.B. JPEG color image encryption method based on high dimensional chaotic system. Laboratory Research and Exploration [J]. 2020, 39(5):136-139. (Chinese)

[2] Wu C.P., Kuo Jay C.C. Design of integrated multimedia compression and encryption systems[J]. IEEE Transactions on Multimedia 2005, 7(5):828-839.

[3] Lian S. Efficient image or video encryption based on spatiotemporal chaos system[J]. Chaos, Solitons & Fractals, 2009, 40(5):2509-2519.

[4] Guo J.W., Zhang D.X., Yang S.S.. Image encryption algorithm combined with JPEG compression. Computer applications and software[J], 2019, 36(5):178-182. (Chinese)

[5] C.Cahit, E. Solak. Cryptanalysis of a chaos-based image encryption algorithm. Physics Letters A[J], 2009, 373: 1357-1360.

[6] Arroyo D., Li C. Q., Li S. J., et al. Cryptanalysis of an image encryption scheme based on a new total shuffling algorithm. Chaos, Solitons and Fractals[J], 2009, 41: 2613-2616.

[7] Fei P., Qiu S. S., L. Min. An image encryption algorithm based on mixed chaotic dynamic systems and external keys. Circuits and Systems [J], 2005: 1135-1139.

[8] Lu Y., Yang W. , Chen L., Encryption algorithm for the image in the frequency domain. Computer Engineering and Application [J], 2003, 39 (14) : 130-131.

[9] Han D., Wang C.H., The encryption algorithm of JPEG image based on DC coefficient hiding. Microelectronics and computers [J], 2018, 35(9):47-51. (Chinese)

[10] Xue X.L., Jin H.Y., Zhou D.S., Zhou C.J., Medical Image Protection Algorithm Based on Deoxyribonucleic Acid Chain of Dynamic Length. Frontiers in Genetics, 2021. DOI: 10.3389/FGENE.2021.654663.