A Quantum Image Encryption Method Based on DNACNot

SHIHUA ZHOU

Key Laboratory of Advanced Design and Intelligent Computing, Ministry of Education, School of Software Engineering, Dalian University, Dalian 116622, China

e-mail: zhoushihua@dlu.edu.cn

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFC0910500, in part by the National Natural Science Foundation of China under Grant 61425002, Grant 61751203, Grant 61772100, Grant 61972266, Grant 61802040, Grant 61172121, and Grant 61572093; in part by the Program for Changjiang Scholars and Innovative Research Team in University under Grant IRT_15R07; in part by the Program for Liaoning Innovative Research Team in University under Grant LT2017012, in part by the Natural Science Foundation of Liaoning Province under Grant 20180551241 and Grant 2019-ZD-0567, in part by the High-Level Talent Innovation Support Program of Dalian City under Grant 2017RQ060 and Grant 2018RQ075, in part by the Dalian Outstanding Young Science and Technology Talent Support Program under Grant 2017RJ08, and in part by the Scientific Research Fund of Liaoning Provincial Education Department under Grant JYJT19051.

ABSTRACT The image is one of the most important means to store information, and is widely used in every aspect of life. However, the characteristics of images enable them to be easily stolen, tampered with, and copied. Researchers have proposed many encryption methods, with most applying only to traditional digital images. There are few encryption methods for quantum images, and quantum image-oriented encryption technology has begun to attract researchers’ attention. This study proposes a quantum image encryption method based on DNA Controlled-Not (DNACNot). Based on the quantum image information, encryption parameters are obtained and transferred as part of a chaotic initial key. Two natural DNA sequences are amplified to obtain the DNA coordinate sequence, and a modified chaos game representation is used to modify the DNA coordinate sequence, which is then used to correct the sequence generated by chaos. The resulting sequence is converted to an integer sequence to obtain DNACNot. A controlled-not operation is performed between DNACNot and the quantum image is scrambled by bit-plane to obtain the encrypted image. The encryption method has high security, a good encryption effect, and a large key space. The method can effectively resist exhaustive, statistical, and differential attacks. The algorithm is easy to implement at a low cost. The encryption time of our proposed method is satisfactory, and the method is suitable for real-time encryption. Moreover, the encryption results can be transmitted over the internet and stored in the cloud.

INDEX TERMS Quantum image, encryption, DNA sequence.

I. INTRODUCTION

Because images are vivid, they are widely used by humans and are an important means to express information. However, they also provide a channel for criminals to use the internet to obtain unauthorized data. Image owners need reliable encryption to protect their interests. Image encryption methods can be based on modern cryptography technology [1], secret segmentation and sharing technology, chaotic technology [2], mathematical formulas or transforms [3], compressive sensing [4], and DNA technology [5]. The latter four methods are the most widely used, and these are the focus of researchers.

Because of the characteristic of chaos, image encryption based on chaos is one of the most popular methods.

Wang et al. [2] proposed a chaotic system in the form of the power-exponential structure, and a color image cryptosystem was proposed based on this system. Encryption methods based on various mathematical formulas or transformations have also received extensive attention. Wu et al. [3] made use of elliptic curve ElGamal encryption to encrypt compressed images. Researchers have also begun to focus on compressive sensing image encryption methods. For example, Hu et al. [4] proposed a method based on double random phase encoding and parallel compressive sensing with updated sampling processes. Another new method [5] is based on DNA computing and coding [6], [7]. Hu et al. [8] proposed a method based on high-dimensional chaotic systems and the cycle operation for DNA sequences, in which pixels of original images were encoded randomly using a DNA coding rule.
Some new methods are neural network-based [9] or asymmetric-based [10]. Almost all of these methods are designed for traditional digital images, while the encryption of quantum images remains in its infancy. However, as the traditional computer approaches its physical limits, researchers are beginning to study a new computing model, and quantum computers have produced a series of satisfactory encryption results [11]–[23]. For example, EI-Latif et al. [11] presented an encryption algorithm of quantum medical images. The proposed algorithm utilizes gray code and a chaotic map. Liu et al. [12] proposed quantum block image encryption based on Arnold transform and Sine Chaotification Model. Hu et al. [14] proposed a quantum image encryption algorithm based on Arnold scrambling and wavelet transforms. Abd-EI-Atty and EI-Latif [16] proposed an encryption protocol for NEQR images based on one-particle quantum walks on a circle. A Double image encryption using 3D Lorenz chaotic system, 2D non-separable linear canonical transform and QR decomposition was proposed by Rakheja et al. [19]. Liu et al. [20] proposed a quantum image encryption algorithm based on bit-plane permutation and sine logistic map. The current encryption technology for quantum images is divided into three categories: scrambling technology based on the basic gray code and bit plane (GB); scrambling technology based on improved GB; and scrambling technology combined with GB and position. These quantum image encryption techniques can only complete one of the two basic requirements of image encryption, that is, scrambling. For the other basic requirement, image encryption–diffusion, the effect need to be improved further.

This study presents a quantum image encryption method based on DNA Controlled-Not (DNACNot). Four parameters are obtained from quantum image information. Based on two natural DNA sequences and the encryption parameters, the DNACNot is obtained by the improved chaotic game representation and iterative chaotic map with infinite collapses (ICMIC). The encrypted image is obtained by a controlled-not operation. The encryption results are analyzed comprehensively, from which it can be seen that the method has high security. Our method greatly improves the diffusivity while preserving the good scrambling of traditional quantum image encryption.

II. BACKGROUND KNOWLEDGE

A. NOVEL ENHANCED QUANTUM REPRESENTATION

In 2013, Zhang et al. [24] proposed novel enhanced quantum representation (NEQR), based on the flexible representation of quantum images (FRQI) model, to represent quantum images. Although NEQR uses more qubits than some other models to represent an image, it can contain both the position and color information of every pixel. The NEQR model for a $2^n \times 2^n$ gray image is

$$|A\rangle = \frac{1}{2^n} \sum_{a=0}^{2^{2n}-1} |c_a\rangle \otimes |a\rangle,$$

where $|c_a\rangle = |c_a^7 \ldots c_a^1 c_a^0\rangle$, $c_a^k \in \{0,1\}, k = \{0,\ldots,7\}$.

For a $128 \times 128$ gray test Lena image (Figure 1), its NEQR is written as follows:

$$|\text{Lena}\rangle = \frac{1}{2^7} (|01110101\rangle \otimes |00000000000000\rangle + \ldots + |10010111\rangle \otimes |11111111111111\rangle).$$

B. BIT-PLANE

The set of bits for each pixel value of an image corresponding to a specific bit position is called a bit plane [11], [12]. For example, if the pixel value range for an image is [0,255], then the binary form represents it as 8-bit data; hence, the image has eight bit planes. Bit-plane 8 contains the most significant bits, and Bit-plane 1 contains the least important bits (Figure 2).

C. CHAOS GAME REPRESENTATION

Jeffrey [25] proposed the scale-independent chaos game representation (CGR) of genomic sequences in 1990. The CGR of a DNA sequence is drawn in a unit square, with its four vertices marked by nucleotides A-(0,0), C-(0,1), G-(1,1), and T-(1,0). The mapping process has the following steps: the first nucleotide of the sequence is drawn between the center of the square and the vertex representing the nucleotide; successive nucleotides in a sequence are drawn between the previously drawn points and the vertices representing the nucleotides to be drawn. The coordinates of the continuous points in the
chaos game representation of DNA sequences are described by an iterative function system defined as
\[\begin{align*}
x_i &= \frac{1}{2}(x_{i-1} + g_{ix}), \\
y_i &= \frac{1}{2}(y_{i-1} + g_{iy}),
\end{align*}\]
(2)
where \(x_i\) and \(y_i\) are the coordinates of the \(ith\) base, and \(g_{ix}\) and \(g_{iy}\) are the vertex values of the \(ith\) base.

**D. ICMIC**
The iterative chaotic map with infinite collapses (ICMIC) [26] is a kind of one-dimensional (1-D) chaotic map that has higher Lyapunov exponents. A perfect iterative chaos model can be obtained by properly selecting the parameters, and the sequences it produces can be regarded as better than those from a uniform distribution. It can be written as
\[x_{n+1} = \sin(a x_n),\]
(3)
where \(x \in (-1, 1)\), and \(x \neq 0\), \(n \geq 1\), and \(n\) is an integer, \(a \in (0, +\infty)\).

**III. A QUANTUM IMAGE ENCRYPTION METHOD BASED ON DNACNot**

**A. ENCRYPTION PARAMETERS**
For an encryption method that better resists differential attacks, we design encryption parameters from the original image information. We use the two-dimensional discrete cosine transform (2D-DCT) to calculate four encryption parameters based on the bit plane information of the original image. Eight bit planes are processed by 2D-DCT to obtain eight decimal matrices, namely, \(D_{plane1}, D_{plane2}, D_{plane3}, D_{plane4}, D_{plane5}, D_{plane6}, D_{plane7}\), and \(D_{plane8}\). The encryption parameters \(sub1\), \(sub2\), \(sub3\), and \(sub4\) are obtained as follows:
\[\begin{align*}
sub1 &= \sum_{i=1}^{M} \sum_{j=1}^{N} (abs(d_{p1}(i, j)) + abs(d_{p8}(i, j))) \\
&\quad - \text{floor}(\sum_{i=1}^{M} \sum_{j=1}^{N} (abs(d_{p1}(i, j)) + abs(d_{p8}(i, j)))),
\end{align*}\]
(4)
\[\begin{align*}
sub2 &= \sum_{i=1}^{M} \sum_{j=1}^{N} (abs(d_{p2}(i, j)) + abs(d_{p7}(i, j))) \\
&\quad - \text{floor}(\sum_{i=1}^{M} \sum_{j=1}^{N} (abs(d_{p2}(i, j)) + abs(d_{p7}(i, j)))),
\end{align*}\]
(5)
\[\begin{align*}
sub3 &= \sum_{i=1}^{M} \sum_{j=1}^{N} (abs(d_{p3}(i, j)) + abs(d_{p6}(i, j))) \\
&\quad - \text{floor}(\sum_{i=1}^{M} \sum_{j=1}^{N} (abs(d_{p3}(i, j)) + abs(d_{p6}(i, j)))),
\end{align*}\]
(6)
\[\begin{align*}
sub4 &= \sum_{i=1}^{M} \sum_{j=1}^{N} (abs(d_{p4}(i, j)) + abs(d_{p5}(i, j))) \\
&\quad - \text{floor}(\sum_{i=1}^{M} \sum_{j=1}^{N} (abs(d_{p4}(i, j)) + abs(d_{p5}(i, j)))),
\end{align*}\]
(7)
where \(d_{p1}(i, j), d_{p2}(i, j), d_{p3}(i, j), d_{p4}(i, j), d_{p5}(i, j), d_{p6}(i, j), d_{p7}(i, j), \) and \(d_{p8}(i, j)\) are the values of the \(ith\) row and the \(jth\) column in \(D_{plane1}\), \(D_{plane2}\), \(D_{plane3}\), \(D_{plane4}\), \(D_{plane5}\), \(D_{plane6}\), \(D_{plane7}\), and \(D_{plane8}\), respectively; \(abs()\) is the absolute value operation; and \(\text{floor()}\) is an integer operation.

**B. IMPROVED CHAOS GAME REPRESENTATION**
We use two natural DNA sequences as encryption tools to further expand the key space and enable the encryption algorithm to better resist exhaustive attacks. The chaos game representation is improved as follows:
\[\begin{align*}
x_i &= \frac{1}{2}x_{i-1} + \frac{1}{4}(e1x_i + e2x_i) \\
y_i &= \frac{1}{2}y_{i-1} + \frac{1}{4}(e1y_i + e2y_i),
\end{align*}\]
(8)
where \(e1x \) and \(e1y\) are the vertex values of one DNA sequence, \(e2x\) and \(e2y\) are the vertex values of another DNA sequence, and \(x\) and \(y\) are the coordinates of the base in the two-dimensional plane.

**C. DNACNot**
DNACNot is constructed by some operations on two natural DNA sequences and chaotic sequences, as follows.

Step 1: Calculate the distances between the chaotic game coordinate points and the origin, and convert the coordinate sequence \(CG\) to the modification sequence \(MOTDNA = \{mDNA_q\}, mDNA_q \in [0, 1], q = \{1, \ldots, 2^{2n}\}\).

Step 2: Set system parameter \(a\) and modification parameter \(subz\), and combine them with encryption parameters to obtain four chaotic sequences of length \(2^{2n}\).

Step 3: Use the sequence \(MOTDNA\) to modify four chaotic sequences in different ways to obtain modified sequences of length \(2^{2n}\).

Step 4: Convert the above sequences to integer sequences of length \(2^{2n}\). Perform XOR operations in turn to obtain \(DNALOGIN\).

Step 5: Represent the integer sequence \(DNALOGIN\) by NEQR to obtain \(DNACNot\).

**D. ALGORITHM DESCRIPTION**
The algorithm has four parts. Four encryption parameters are obtained and transferred as part of the key based on the quantum image information. Two natural DNA sequences are amplified to obtain the DNA coordinate sequence, which is modified by chaos game representation. The modified sequence corrects the sequences generated by ICMIC, and the result is converted to an integer sequence to obtain...
DNACNot. A controlled-not operation is used to calculate DNACNot and the quantum image is scrambled by bit-plane to obtain the encrypted image. A flowchart illustrating the process is shown in Figure 3, and the encryption approach is as follows.

Step 1: Input a quantum image $|A\rangle$ of size $2^n \times 2^n$, represented by NEQR.

Step 2: Obtain four encryption parameters, sub1, sub2, sub3, and sub4, based on the quantum image $|A\rangle$.

Step 3: Obtain two natural DNA sequences DNA1 and DNA2 by the given parameters. These are amplified to obtain the new DNA sequences ENDNA1 and ENDNA2, of length $2^{2n}$.

Step 4: Convert ENDNA1 and ENDNA2 to vertex sequences EDNAT1 and EDNAT2 of length $2^{2n}$.

Step 5: Given $(x_0, y_0)$, use the improved chaos game representation to convert vertex sequences EDNAT1 and EDNAT2 to one chaotic game coordinate sequence CG, thus obtaining the modification sequence MODNA.

Step 6: Select parameters $a$ and $subz$, combined with encryption parameters sub1, sub2, sub3, and sub4, to obtain chaotic sequences LOGOR1, LOGOR2, LOGOR3, and LOGOR4, respectively, all of length $2^{2n}$.

Step 7: Use modification sequence MODNA to modify chaotic sequences LOGOR1, LOGOR2, LOGOR3, and LOGOR4 in different ways, to obtain modified sequences DNALOG1, DNALOG2, DNALOG3, and DNALOG4, respectively, all of length $2^{2n}$.

Step 8: Convert sequences DNALOG1, DNALOG2, DNALOG3, and DNALOG4 to respective integer sequences DNALOGIN1, DNALOGIN2, DNALOGIN3, and DNALOGIN4, all of length $2^{2n}$. The method for obtaining DNALOGIN is as follows: after DNALOGIN1 and DNALOGIN2 are XORed, the result is XORed with DNALOGIN3, and this result is XORed with DNALOGIN4.

Step 9: Represent integer sequence DNALOGIN by NEQR to get $|DNACNot\rangle$.

Step 10: Scramble quantum image $|A\rangle$ using bit plane technology to obtain the scrambled quantum image $|S\rangle$.

Step 11: The quantum image $|S\rangle$ and $|DNACNot\rangle$ undergo the controlled-not operation to obtain the encrypted quantum image $|E\rangle$.

Decryption follows the same steps and proceeds by executing steps 11 and 10 in reverse. The decryption steps are as follows.

Step 1: Follow the same steps as the encryption algorithm to generate $|DNACNot\rangle$.

Step 2: The encrypted quantum image $|E\rangle$ and $|DNACNot\rangle$ undergo the controlled-not operation to obtain the quantum image $|S\rangle$.

Step 3: Use reserve bit plane technology to scramble quantum image $|S\rangle$, and then obtain the original quantum image $|A\rangle$.

**IV. EXPERIMENTAL RESULTS**

Three simulation experiments were implemented on MATLAB 2017a. The quantum images were of size $128 \times 128$. The keys of every image were $\{(154, 1021, 10), (101, 241, 10), (0.45, -0.12), 2, 0.326264742\}$. The meaning of each part in the secret key set is divided into $(154, 1021, 10)$ and $(101, 241, 10)$, and are used to query two natural DNA sequences on the website www.ncbi.nlm.nih.gov, respectively. Numbers 154 and 101 are the IDs of two sequences, 1021 and 241 are the starting positions, and 10 and 10 are the lengths; $(0.45, -0.12)$ represents the starting coordinate position for the improved chaos game representation; 2 is the system parameter of ICMIC; and 0.54319847326264742 is...
FIGURE 5. Other experimental images.

used to modify the four encryption parameters. Experimental results are shown in Figure 4. Figure 4(a)–(c) show the plain images; Figure 4(d)–(f) show the encrypted images; Figure 4(g)–(i) show the decrypted images.

To effectively compare these results with those on other images, we encrypted six other images. Figure 5(a) (with the size of 128 × 128) is from EI-Latif et al. [11], Figure 5(b) (with the size of 256 × 256) is from Liu et al. [12], Figure 5(c) (with the size of 256 × 256) is from Hu et al. [14], Figure 5(d) (with the size of 256 × 256) is from Abd-El-Atty and EI-Latif [16], Figure 5(e) (with the size of 256 × 256) is from Rakheja et al. [19], and Figure 5(f) (with the size of 256 × 256) is from Liu et al. [20].

V. ALGORITHM PERFORMANCE ANALYSIS
A. KEY SPACE ANALYSIS
This encryption method has 10 key parameters, and it uses the natural DNA sequence as the key. Although there are only four types of bases, natural DNA sequences are diverse due to different combinations and lengths. Even if the same DNA sequence is selected at a different starting position, it is possible to obtain completely different subsequences. Moreover, with the continuous development of genetic engineering and sequencing technology, the known DNA sequences of human beings are increasing, and people can treat the natural DNA sequence set as a one-time pad. Therefore, the encryption method has the property of a one-time pad. It also uses the improved chaotic game representation method and ICMIC, whose parameters further expand the key space, thus enabling the proposed quantum image encryption method to well resist exhaustive attacks.

B. KEY SENSITIVITY ANALYSIS
A secure encryption scheme is sensitive to subtle changes in the key. To test the key sensitivity of the proposed method, three new keys were used to decrypt the original encrypted image (Figure 4(d)). Each new key differed only slightly from the original, where key1 = \{154, 1020, 10\}, \{101, 241, 10\}, \{(0.45, -0.12), 2, 0.326264742\}, key2 = \{154, 1021, 10\}, \{101, 241, 10\}, \{(0.45, -0.121), 2, 0.326264742\}, and key3 = \{154, 1021, 10\}, \{101, 241, 10\}, \{(0.45, -0.12), 2, 0.326264743\}.

The experimental results obtained when using the wrong keys are shown in Figure 6. Figure 6(a)–(c) show the decrypted results under key1, key2, and key3, respectively. From Figure 6 and Table 1, it can be seen that the original image cannot be decrypted with the wrong keys. Thus, this method has satisfactory key sensitivity.

C. CORRELATION ANALYSIS
A secure image encryption scheme must eliminate the strong correlation between pixels of an image. Correlation coefficients were used to analyze the correlation between pixels. The correlation coefficient between pixel x and pixel y is

$$r_{xy} = \frac{E((x - E(x))(y - E(y)))}{\sqrt{D(x)D(y)}},$$

$$E(x) = \frac{1}{N} \sum_{i=1}^{N} x_i,$$

where $E(x)$ and $D(x)$ are the expectation and variance, respectively, and $r_{xy} \in [-1, 1]$. 

FIGURE 6. Experimental results obtained when using the wrong keys.

TABLE 1. NPCR of decrypted images.

|        | Figure 6(a) | Figure 6(b) | Figure 6(c) |
|--------|-------------|-------------|-------------|
| NPCR   | 0.9962      | 0.9968      | 0.9962      |
Figure 7(a)–(c) show the correlation among pixels of the plain images (Figure 4(a)–(c)) in the vertical direction, and Figure 7(d)–(f) show the correlation of their encrypted images (Figure 4(d)–(f)) in the same direction. Figure 7 and Table 2 show that the correlation of the original images is very high, whereas the correlation of the encrypted images is very low. The closer the correlation between adjacent pixels is to 0, the greater the difference between them, and the more difficult it is for an attacker to use the relationship between pixels to infer the pixel values of other pixels. The absolute values of our results are very close to 0, and thus the attacker cannot get the information of the original image through the relationship between pixels. For convenience of comparison, this method is used to encrypt Figure 5(a)–(f). The results are shown in Table 3. We encrypt the same images as those in the literature, and the results we obtain are closer to 0 compared with their results. Thus, our encrypted images have higher security.

### TABLE 2. Correlation coefficients of plain and encrypted images.

|                  | Horizontal | Vertical | Diagonal |
|------------------|------------|----------|----------|
| Figure 4(a)      | 0.9530     | 0.9711   | 0.9422   |
| Figure 4(d)      | 0.0037     | 0.0911   | 0.0004   |
| Figure 4(b)      | 0.8999     | 0.9331   | 0.8690   |
| Figure 4(e)      | 0.0167     | 0.0020   | 0.0039   |
| Figure 4(c)      | 0.8385     | 0.9357   | 0.8458   |
| Figure 4(f)      | 0.0079     | 0.0093   | -0.0127  |

### TABLE 3. Comparison of correlation coefficients.

|                  | Horizontal | Vertical | Diagonal |
|------------------|------------|----------|----------|
| Figure 5(a) [11] | 0.9744     | 0.9760   | 0.9550   |
| EL-Latif [11]    | -0.0020    | -0.0095  | -0.0015  |
| Ours             | 0.0019     | 0.0101   | 0.0009   |
| Figure 5(b) [12] | 0.5776     | 0.6725   | 0.6012   |
| Liu [12]         | -0.0423    | 0.0202   | -0.0212  |
| Ours             | -0.0124    | 0.0168   | -0.0130  |
| Figure 5(c) [14] | 0.9455     | 0.9314   | 0.9148   |
| Hu [14]          | 0.0062     | -0.0217  | 0.0002   |
| Ours             | -0.0087    | 0.0098   | 0.0030   |
| Figure 5(d) [16] | 0.9555     | 0.9241   | 0.8990   |
| Abd-El-Atty [16]| -0.0126    | 0.0002   | -0.0047  |
| Ours             | 0.0031     | 0.0010   | -0.0019  |
| Figure 5(f) [20] | 0.9507     | 0.9281   | 0.8713   |
| Liu [20]         | 0.0153     | 0.0191   | 0.0055   |
| Ours             | 0.0058     | 0.0071   | 0.0051   |

D. HISTOGRAM ANALYSIS

A secure image encryption algorithm can resist statistical attacks. It is necessary to eliminate the correlation between pixels and ensure that the pixels of an encrypted image are evenly distributed. We performed a histogram analysis to verify that the proposed method can evenly distribute pixels. The variance of the histogram is

\[
\text{var}(x) = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{1}{2} (x_i - x_j)^2.
\]  

(11)

Table 4 shows the \( \text{var} \) of Figure 4, and Figure 8(a)–(f) are the histograms corresponding to Figure 4(a)–(f), respectively.
We can see that $var$ of the original images are all very high while the $var$ of the encrypted images are all very low. Table 4 shows that the $var$ after encryption is reduced by at least 97% compared to before encryption. The smaller the $var$, the more even the pixel value distribution of the pixels; this makes it more difficult for an attacker to use the distribution of pixel values to crack an encrypted image. Figure 8 also illustrates this conclusion. Figure 8(a)–(c) are histograms of the original images. They all contain very obvious crests, and their fluctuations are very obvious. Attackers can use the histogram information to speculate about and analyze image information. Figure 8(d)–(f) are histograms of the encrypted images (Figure 4(a)–(f)). The information entropies of Figure 4(d)–(f) are very close to the theoretical values. Table 7 shows that the information entropies of the plain images are all very low. The smaller the $var$, the more even the pixel value distribution of the pixels; this makes it more difficult for an attacker to use the distribution of pixel values to crack an encrypted image. Figure 8 also illustrates this conclusion. Figure 8(a)–(c) are histograms of the original images. They all contain very obvious crests, and their fluctuations are very obvious. Attackers can use the histogram information to speculate about and analyze image information. Figure 8(d)–(f) are histograms of the encrypted images (Figure 4(a)–(f)). The information entropies of Figure 4(d)–(f) are very close to the theoretical values. Table 6 shows that the information entropies of the plain and encrypted images (Figure 4(a)–(f)). The information entropies of Figure 4(d)–(f) are very close to the theoretical values. Table 7 shows the comparison results between our method and others’ in the case of encrypting the same images. The values obtained by our method are closer to 8 than those from EI-Latif et al. [16], Abd-El-Atty et al. [17], and Rakheja et al. [19]. Although our result is not better than that of Liu et al. [20], the results differ by only 0.0001. Therefore, in terms of information entropy, the performance of our algorithm is satisfactory.

**F. DIFFERENTIAL ATTACK**

A differential attack cracks the symmetric encryption scheme by analyzing the information distribution of an encrypted image. A secure symmetric encryption scheme can resist a differential attack. The number of pixels change rate (NPCR) and unified average changing intensity (UACI) are used to measure the ability to resist differential attacks.

\[
C(i,j) = \begin{cases} 
0, & \text{if } T_1(i,j) = T_2(i,j) \\
1, & \text{if } T_1(i,j) \neq T_2(i,j) 
\end{cases} 
\]  
\[\text{NPCR} = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} C(i,j)}{m \times n} \times 100\%.
\]
\[\text{UACI} = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} |T_1(i,j) - T_2(i,j)|}{255 \times m \times n} \times 100\%.
\]

**TABLE 4. Analysis of histogram variance.**

| Image       | Histogram variance |
|-------------|--------------------|
| Figure 4(a) | 11754.3875         |
| Figure 4(d) | 213.8750           |
| Figure 4(b) | 9094.3125          |
| Figure 4(c) | 227.7183           |
| Figure 4(e) | 17389.3750         |
| Figure 4(f) | 242.3750           |

**TABLE 5. Comparison of variance of histograms.**

|        | Original | Others’ | Ours  |
|--------|----------|---------|-------|
| Figure 5(b) | 65972.5468 [12] | 269.3906 [12] | 254.8438 |
| Figure 5(f) | 34877.968 [20] | 243.6953 [20] | 239.4871 |

**TABLE 6. Analysis of information entropy.**

| Image       | Information entropy |
|-------------|---------------------|
| Figure 4(a) | 7.3671              |
| Figure 4(d) | 7.9906              |
| Figure 4(b) | 7.5422              |
| Figure 4(e) | 7.9900              |
| Figure 4(c) | 7.1279              |
| Figure 4(f) | 7.9894              |

**TABLE 7. Comparison of information entropy.**

|        | Original | Others’ | Ours  |
|--------|----------|---------|-------|
| Figure 5(a) | 7.1024[11] | 7.9878[11] | 7.9896 |
| Figure 5(b) | 7.1273[12] | 7.9970[12] | 7.9972 |
| Figure 5(d) | 6.9046[16] | 7.9967[16] | 7.9970 |
| Figure 5(e) | 7.5893[19] | 7.9958[19] | 7.9970 |
| Figure 5(f) | 7.5659[20] | 7.9973[20] | 7.9972 |

**TABLE 8. NPCR and UACI tests.**

|        | 0.05-level | 0.01-level | 0.001-level |
|--------|------------|------------|-------------|
| Theoretical NPCR | 99.525% | 99.496% | 99.458% |
| Figure 4(a) (99.57%) | Pass | Pass | Pass |
| Figure 4(b) (99.55%) | Pass | Pass | Pass |
| Figure 4(c) (99.57%) | Pass | Pass | Pass |
| Theoretical UACI | 33.101% | 32.987% | 32.855% |
| Figure 4(a) (33.41%) | Pass | Pass | Pass |
| Figure 4(b) (33.47%) | Pass | Pass | Pass |
| Figure 4(c) (33.51%) | Pass | Pass | Pass |

Table 6 shows the information entropies of the plain and encrypted images (Figure 4(a)–(f)). The information entropies of Figure 4(d)–(f) are very close to the theoretical values. Table 7 shows the comparison results between our information entropies and others’ in the case of encrypting the same images. The values obtained by our method are closer to 8 than those from EI-Latif et al. [11], Liu et al. [12], Abd-El-Atty et al. [16], and Rakheja et al. [19]. Although our result is not better than that of Liu et al. [20], the results differ by only 0.0001. Therefore, in terms of information entropy, the performance of our algorithm is satisfactory.
TABLE 9. Comparison results for different image encryption algorithms on NPCR.

| Algorithm          | NPCR (128 × 128) | NPCR (256 × 256) | NPCR (512 × 512) | NPCR (1024 × 1024) |
|--------------------|------------------|------------------|------------------|-------------------|
| Theoretical value  | 99.3292%         | 99.4960%         | 99.4588%         |                   |
| EI-Latif et al. [11] (99.643%) | Pass | Pass | Pass |                   |
| Ours (99.602%)     | Pass | Pass | Pass |                   |
| Theoretical value  | 99.5693%         | 99.5527%         | 99.5341%         |                   |
| Liu [12] (99.559%) | Fail | Fail | Fail |                   |
| Ours (99.6313%)    | Pass | Pass | Pass |                   |
| Abd-El-Atty [16] (99.58%) | Pass | Pass | Pass |                   |
| Ours (99.6115%)    | Pass | Pass | Pass |                   |
| Liu [20] (99.5300%) | Fail | Fail | Fail |                   |
| Ours (99.5901%)    | Pass | Pass | Pass |                   |

TABLE 10. Comparison results for different image encryption algorithms on UACI.

| Algorithm          | UACI (128 × 128) | UACI (256 × 256) | UACI (512 × 512) | UACI (1024 × 1024) |
|--------------------|------------------|------------------|------------------|-------------------|
| Theoretical value  | 33.1012%         | 32.9874%         | 32.9532%         |                   |
| EI-Latif et al. [11] (28.9754%) | Fail | Fail | Fail |                   |
| Ours (33.1797%)    | Pass | Pass | Pass |                   |
| Theoretical value  | 33.2824%         | 33.2255%         | 33.1594%         |                   |
| Liu [12] (33.6197%) | Pass | Pass | Pass |                   |
| Ours (33.39%)      | Pass | Pass | Pass |                   |
| Abd-El-Atty [16] (34.76%) | Fail | Fail | Fail |                   |
| Ours (33.6914%)    | Pass | Pass | Pass |                   |
| Liu [20] (33.5291%) | Pass | Pass | Pass |                   |
| Ours (33.5716%)    | Pass | Pass | Pass |                   |

EI-Latif et al. [11] only meets three criteria, and it only meets the criteria of NPCR, and none of the three indicators of UACI meet the criteria. Liu et al. [12] meets five criteria, but it is not good on NPCR at the 0.05-level. The algorithm in Abd-El-Latif et al. [16] only meets three criteria, and it fails on UACI at three levels. Although the algorithm in Liu et al. [20] can meet all criteria on UACI, it cannot meet any criterion on NPCR. In the case of encrypting the same images, our proposed algorithm meets all criteria. These findings further prove that our method can effectively resist differential attacks compared to other algorithms.

G. COMPUTATIONAL COMPLEXITY AND ENCRYPTION SPEED ANALYSIS

According to Chai et al. [28], the time complexities on the preparation and recovery of the quantum image for NEQR representation are $O(qn2^{2n})$ and $O(2^{2n})$, respectively. However, generally speaking, the time complexity of the quantum image encryption algorithm only includes the time cost in the encryption process or the decryption process, and does not include the time cost in the preparation and recovery. Therefore, the time cost in the preparation and recovery of the quantum image is neglected. In our method, the time complexity consists of four parts: the time cost of generating key parameters, the time cost of processing the natural DNA sequence, the time cost of generating DNACnot, and the time cost of a controlled-not operation. In the stage of generating key parameters, the complexity is $O(n^2)$; in the stage of processing the natural DNA sequence, the complexity is $O(2^{2n})$; in the stage of processing the natural DNA sequence, the complexity is $O(2^{2n})$; and in the stage of a controlled-not operation, the complexity is $O(2^{2n})$. Therefore, the computational complexity of our quantum image encryption method is $O(2^{2n})$, which is as good as that of classical methods [20].

For a good encryption algorithm, the speed of encryption is also very important. In order to more intuitively introduce the speed of the quantum image encryption method we proposed, we used multiple images with different sizes but with the same image content as Figure 4(a)-(c) to record encryption time required for them. We used MATLAB 2019 to simulate the experiments. The configuration of the computer we used is: Intel Core i7-9700k CPU, 16GB memory and Windows 10 operation system. The results are shown in Table 11. For images with the size of 128 × 128, all encryption times are less than 0.013 seconds; for images with the size of 256 × 256, all encryption times are less than 0.042 seconds; for images with the size of 512 × 512, all encryption times are less than 0.185 seconds; even for large images with the size of 1024 × 1024, all encryption time are less than 0.76 seconds; for super-large images with the size of 2048 × 2048 that are rarely used in normal times, the encryption time only takes about 3 seconds. The encryption time of our proposed method is satisfactory, and the method is suitable for real-time encryption. To facilitate comparison with others’ results, we tested the same images as others’ literature. Table 12 shows the comparison results. It can be seen from Table 12 that the encryption times required by our method are obviously shortened compared to other algorithms.
VI. CONCLUSION

We proposed a quantum image encryption method based on DNACNot. Four encryption parameters obtained from quantum image information were used as the chaotic initial value. Two natural DNA sequences were used according to the improved chaos game representation to modify the chaotic sequences to build DNACNot. A controlled-not operation was performed with DNACNot, and the quantum image was scrambled by a bit plane to obtain the encrypted image. Our algorithm was shown to be effective at encrypting quantum images. Through five aspects of performance analysis, namely, key space, key sensitivity, correlation, histograms, and differential attacks, we proved that our encryption method has high security. To further prove the superiority of our encryption method, under the premise of encrypting the same original images, we compared our experimental results with those of others. Through comparison, we can see that in each performance analysis our method is overall better than others. Although in terms of information entropy our result is not superior to Liu et al. [20], the difference is only 0.0001. Except for this, our method performs better than that in Liu et al. [20]. Therefore, regarding overall performance, our method is also superior to the method in Liu et al. [20]. In addition to the above performance analysis, we also analyzed the speed and computational complexity of the encryption algorithm. These two indicators are also satisfactory. In subsequent research, we will further optimize the structure of the encryption algorithm, improve its efficiency, reduce its computational complexity, and improve the security of encryption.

REFERENCES

[1] S. M. Wadi and N. Zainal, “Rapid encryption method based on AES algorithm for grey scale HD image encryption,” Procedia Technol., vol. 11, pp. 51–56, Jan. 2013.
[2] M. Wang, X. Wang, Y. Zhang, S. Zhou, T. Zhao, and N. Yao, “A novel chaotic system and its application in a color image cryptosystem,” Opt. Lasers Eng., vol. 121, pp. 479–494, Oct. 2019.
[3] J. Wu, X. Liao, and B. Yang, “Color image encryption based on chaotic systems and elliptic curve ElGamal scheme,” Signal Process., vol. 141, pp. 109–124, Dec. 2017.
[4] G. Hu, D. Xiao, Y. Wang, T. Xiang, and Q. Zhou, “Securing image information using double random phase encoding and parallel compressive sensing with updated sampling processes,” Opt. Lasers Eng., vol. 98, pp. 123–133, Nov. 2017.
[5] X. Chai, X. Fu, Z. Gan, Y. Lu, and Y. Chen, “A color image cryptosystem based on dynamic DNA encryption and chaos,” Signal Process., vol. 155, pp. 44–62, Feb. 2019.
[6] B. Wang, Q. Zhang, and X. Wei, “Tabu variable neighborhood search for designing DNA barcodes,” IEEE Trans. Nanobiosci., vol. 19, no. 1, pp. 127–131, Jan. 2020.
[7] X. Li, B. Wang, H. Lv, Q. Yin, Q. Zhang, and X. Wei, “Constraining DNA sequences with a triplet-bases unpaired,” IEEE Trans. Nanobiosci., vol. 19, no. 2, pp. 299–307, Apr. 2020.
[8] T. Hu, Y. Liu, L.-H. Gong, and C.-J. Ouyang, “An image encryption scheme combining chaos with cycle operation for DNA sequences,” Nonlinear Dyn., vol. 87, no. 1, pp. 51–66, Jan. 2017.
[9] K. Ratnavelu, M. Kalpana, P. Balasubramaniam, K. Wong, and P. Kavitha, “Image encryption method based on chaotic fuzzy cellular neural networks,” Signal Process., vol. 140, pp. 87–96, Nov. 2017.
[10] Z. Wu, X. Zhang, and X. Zhong, “Generalized chaos synchronization circuit simulation and asymmetric image encryption,” IEEE Access, vol. 7, pp. 37898–38008, 2019.
[11] A. A. Abd El-Latif, B. Abd-El-Atty, and M. Talha, “Robust encryption of quantum medical images,” IEEE Access, vol. 6, pp. 1073–1081, 2018.
[12] X. Liu, D. Xiao, W. Huang, and C. Liu, “Quantum block image encryption based on arnold transform and sine chaotic transformation,” IEEE Access, vol. 7, pp. 57188–57199, 2019.
[13] M. Khan and A. Rashid, “Permutation-based special linear transforms with application in quantum image encryption algorithm,” Quantum Inf. Process., vol. 18, no. 10, p. 298, Oct. 2019.
[14] W.-W. Hu, R.-G. Zhou, J. Luo, S.-X. Jiang, and G.-F. Luo, “Quantum image encryption algorithm based on arnold scrambling and wavelet transforms,” Quantum Inf. Process., vol. 19, no. 3, p. 82, Mar. 2020.
[15] N. Jiang, X. Dong, H. Hu, Z. Ji, and W. Zhang, “Quantum image encryption based on henon mapping,” Int. J. Theor. Phys., vol. 58, no. 3, pp. 979–991, Mar. 2019.
[16] B. Abd-El-Atty, A. A. Abd El-Latif, and S. E. Venegas-Andraca, “An encryption protocol for NEQR images based on one-particle quantum walks on a circle,” Quantum Inf. Process., vol. 18, no. 9, p. 272, Sep. 2019.
[17] H. Liu, B. Zhao, and L. Huang, “Quantum image encryption scheme using arnold transform and S-box scrambling,” Entropy, vol. 21, no. 4, p. 343, Mar. 2019.
[18] H.-S. Li, C. Li, X. Chen, and H. Xia, “Quantum image encryption based on phase-shift transform and quantum Haar wavelet packet transform,” Modern Phys. Lett. A, vol. 34, no. 26, Aug. 2019, Art. no. 1950214.
[19] P. Rakheja, R. Vig, and P. Singh, “Double image encryption using 3D lorentz chaotic system, 2D non-separable linear canonical transform and QR decomposition,” Opt. Quantum Electron., vol. 52, no. 2, p. 103, Feb. 2020.
[20] X. Liu, D. Xiao, and C. Liu, “Quantum image encryption algorithm based on bit-plane permutation and sine logistic map,” Quantum Inf. Process., vol. 19, no. 8, p. 239, Aug. 2020.
[21] Y. Luo, S. Tang, J. Liu, L. Cao, and S. Qiu, “Image encryption scheme by combining the hyper-chaotic system with quantum coding,” Opt. Lasers Eng., vol. 124, Jan. 2020, Art. no. 105836.
[22] F. Musanna and S. Kumar, “Image encryption using quantum 3-D baker map and generalized gray code coupled with fractional Chen’s chaotic system,” Quantum Inf. Process., vol. 19, no. 8, p. 19, Aug. 2020.
[23] C. Hou, X. Liu, and S. Feng, “Quantum image scrambling algorithm based on discrete baker map,” Modern Phys. Lett. A, vol. 35, no. 17, Jun. 2020, Art. no. 2050145.
[24] Y. Zhang, K. Lu, Y. Gao, and M. Wang, “NEQR: A novel enhanced quantum representation of digital images,” Quantum Inf. Process., vol. 12, no. 8, pp. 2833–2860, Aug. 2013.
[25] H. J. Jeffrey, “Chaos game representation of gene structure,” Nucleic Acids Res., vol. 18, no. 8, pp. 2163–2170, 1990.
[26] H. C. He, L.-G. Jiang, H.-W. Zhu, and G.-R. Hu, “Chaotic characteristics of a one-dimensional iterative map with infinite collapses,” IEEE Trans. Circuits Syst. I, Fundam. Theory Appl., vol. 48, no. 7, pp. 900–906, Jul. 2001.
[27] Y. Wu, J. Noonan, and S. Agaian, “NPCR and UACI randomness tests for image encryption,” Cyber J., Multidisciplinary J. Sci. Technol., J. Sel. Areas Telecommun., vol. 1, pp. 31–38, Apr. 2011.
[28] X. Chai, Y. Chen, and L. Broyde, “A novel chaos-based image encryption algorithm based on bit-combination scrambling in decimal system and dynamic diffusion,” IEEE Access, vol. 7, pp. 103662–103677, 2019.