IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 16, 2020, accepted September 2, 2020, date of publication September 7, 2020,
date of current version September 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3022398

Color Image Compression-Encryption Using Fractional-Order Hyperchaotic System and DNA Coding

HAO DONG, ENJIAN BAI, XUE-QIN JIANG, AND YUN WU
School of Information Science and Technology, Donghua University, Shanghai 201620, China

Corresponding author: Enjian Bai (baiej@dhu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61772129 and Grant 61972080.

ABSTRACT Recently, many encryption algorithms based on fractional-order chaotic system have been proposed to solve the problem of image encryption. In this paper, we propose a novel color image compression-encryption algorithm based on fractional-order hyperchaotic system and DNA coding. In the data compression stage, the data of R, G, and B channels of the color image are converted to the frequency domain by two-dimensional discrete cosine transform (DCT), and then the amount of encrypted data is reduced by the quantization process. We design a data processing algorithm to ensure that the data after DCT is compatible with DNA coding data format. In the encryption stage, the processes of DNA encoding and decoding, DNA operation, and pixel scrambling are all controlled by the corresponding chaotic sequences, which are generated by the chaotic system. The original image is used to calculate the initial state of the chaotic system, which improves the performance of the algorithm against the chosen-plaintext attack significantly. Experimental results and security analysis illustrate that the proposed algorithm has excellent compression and security performance. It can not only reconstruct the original image well under the condition of low compression ratio, but also provide high security to resist various attacks. Besides, experimental results also indicate that the algorithm proposed can be applied to the fields of color image compression, encryption, and transmission.

INDEX TERMS Color image encryption, DNA encoding, fractional-order hyperchaotic system, discrete cosine transform(DCT), image compression.

I. INTRODUCTION

One of the important signs of the information age is that people can realize information sharing without the limitation of time and space. The transmitted data also transits from the original text data to the multimedia data, and the color image is an important part of the multimedia data. Although classical cryptography can encrypt one-dimensional (1D) data well [1], it is not suitable for two-dimensional (2D) data such as image [2]–[3]. And because of the larger amount of information carried by color image, it is necessary to design a compression-encryption algorithm which can transmit it efficiently and safely.

Since Lorenz discovered the first chaotic attractor in 1963 [4], the research on the chaotic system has taken root and flourished. Because the state of the chaotic system is disordered and the generated chaotic sequence is highly sensitive to the initial value of the system [5], there are more and more complex chaotic systems [6], synchronization control methods between chaotic systems [7], and image encryption algorithm based on the chaos [8]–[11] have been proposed. Hua *et al.* used a 2D Logistic-adjusted-Sine map to encrypt images [9]. In [10], Pak *et al.* used the combination of the 1D chaotic map to encrypt color images. Zhang *et al.* introduced the unified image encryption algorithm based on chaos and cubic S-Box [11]. However, the trajectory of the low-dimensional chaotic system is simple and can be predicted to a certain extent, and its key space is small. Therefore, high dimensional chaotic systems have been applied to more and more encryption algorithms [12]–[19]. A new approach of image encryption based on 3D chaotic map is proposed through Hossain *et al.* [12]. Zhang *et al.* proposed a color image encryption algorithm based on the spatiotemporal chaos of the nonlinear coupled map lattices

The associate editor coordinating the review of this manuscript and approving it for publication was Sun Junwei.

and genetic operations [13]. Ye *et al.* designed a block chaotic image encryption scheme based on self-adaptive modelling [17]. Yang *et al.* used Qi hyperchaotic system and singular value decomposition in YCbCr color space to encrypt color images [18]. Although the image encryption algorithm designed by high-dimensional chaotic system improves its security performance greatly, it is not involved in image compression.

Due to the high correlation and redundancy of image pixels, how to compress the image for transmission has become an urgent problem. To solve this problem, many related algorithms are proposed [20]–[26]. Chai *et al.* used block compressive sensing (CS) and elementary cellular automata to encrypt and compress images [20]. Li *et al.* introduced joint image encryption and compression schemes based on 16 × 16 DCT [21]. Gong *et al.* designed an image compression-encryption scheme by combining the hyperchaotic system with discrete fractional random transform [25].

DNA coding has the characteristics of high parallelism, low power consumption and high information density [27]. The application of DNA coding and DNA computing based on the chaotic system can further enhance the effect of pixel scrambling and confusion, and improve the security and parallelism of the algorithm [28]–[40]. Liu *et al.* proposed an RGB image encryption algorithm based on DNA encoding and chaos maps [28]. A novel chaotic image encryption scheme based on DNA sequence operations is proposed through Wang *et al.* [29]. Wang *et al.* proposed a novel color image encryption scheme based on DNA permutation and the Lorenz system [31]. Wu *et al.* used a 2D Hénon-Sine map and DNA approach to encrypt images [33]. Liu *et al.* presented an image encryption scheme based on the hyperchaotic system and DNA with fixed secret keys [38].

With the deepening of research, people find that there are a lot of fractional dimensions in reality, and fractional dimensions can describe natural phenomena preferably compared with the traditional integer-order chaotic system [41]. Besides, the fractional-order chaotic system has the characteristics of nonlocality, high nonlinearity, and can greatly enhance the key space [42], which makes the research on it become a new trend. Although the solution of fractional-order differential equation is more complex than that of integer order, many papers have proposed based on the Adomian decomposition method (ADM) to solve this problem [43]–[47], which can effectively reduce the calculation time. Then, more efficient image encryption schemes based on the fractional-order chaotic system have been proposed [48]–[59]. Yang *et al.* designed an image compression-encryption scheme based on fractional-order hyperchaotic systems combined with 2D compressed sensing and DNA encoding [49]. In [55], the authors proposed a three-dimensional fractional-order discrete Hopfield neural network and applied it to image encryption. In [56], Ismail *et al.* presented a lossless image encryption algorithm based on edge detection and generalized chaotic maps for key generation. Generalized chaotic maps are used to design pseudo-random number key generator. In [58], the authors proposed an image compression-encryption scheme based on set partitioning in hierarchical trees coding-decoding algorithm and synchronization of chaotic maps with non-integer order. Although the image encryption algorithm based on the chaotic system develops rapidly, not all algorithms based on this principle are secure. For example, image encryption algorithms [10], [28] and [33] are all proved to be able to be cracked by chosen-plaintext attack [60]–[62], because these encryption algorithms are not sensitive to changes of plaintext. Among the recently proposed algorithms, the fractional-order chaotic system is not used in the algorithm [38] and DNA computing is not used in the algorithm [42]. In addition, algorithm [48] only uses DNA addition and subtraction to encrypt images and algorithm [49] can only encrypt the gray image.

According to the above analysis, we design a new image compression-encryption algorithm to overcome these shortcomings, which has the following contributions. First of all, our algorithm can encrypt color images of any size effectively and can compress the image according to different quantization matrix. Secondly, we calculate the initial value of the chaotic system through different bit-planes of the image, which makes the initial value of the chaotic system highly sensitive to the plaintext and improves the performance of the encryption algorithm against the chosen-plaintext attack. Thirdly, we use four kinds of DNA operations (add, subtract, XOR, XNOR) to further confuse image pixels, and these four operations and DNA coding are controlled by chaotic sequence, which greatly enhances its security performance. At last, we give the calculation process of using the Adomain algorithm to solve the fractional-order hyperchaotic Chen system in detail, and take the initial state of the fractional chaotic system as the key, which greatly improves the key space of the system.

The rest of this paper is arranged as follows. Section 2 introduces the preliminaries about the fractional-order hyperchaotic Chen system, the Adomian decomposition method, the solution of fractional-order hyperchaotic Chen system and DNA coding and operations. Section 3 describes the image compression-encryption algorithm based on fractional-order hyperchaotic Chen system and DCT in detail. Section 4 deals with the experimental simulation and security analysis. Finally, a brief conclusion is drawn in Section 5.

II. PRELIMINARIES

A. FRACTIONAL-ORDER HYPERCHAOTIC CHEN SYSTEM

In 1999, Chen first proposed Chen chaotic system [63]. Then the hyperchaotic Chen system is proposed, which is defined as follows [64]:

$$\begin{cases} \dot{x_1} = a(x_2 - x_1) + x_4 \\ \dot{x_2} = bx_1 + x_1x_3 + cx_2 \\ \dot{x_3} = x_1x_2 - dx_3 \\ \dot{x_4} = x_2x_3 + rx_4, \end{cases} \tag{1}$$

where $\dot{x_i}$ are the derivative of system state variable $x_i (i = 1, 2, 3, 4)$. a, b, c, d and r are positive parameters. When $a = 35, b = 3, c = 12, d = 7, r \in (0.1085, 0.1798]$, the system of Eq. (1) is hyperchaotic.

In this paper, the Adomian decomposition method is used to solve the fractional differential equation, which uses Caputo differential operator. Caputo fractional differential is defined as follows [65]:

$$ {}^{*}D_t^q f(t) = \begin{cases} 0 \dfrac{1}{\Gamma(m-q)} \int_0^t \dfrac{f(m)(\tau)}{(t-\tau)^{q+1-m}} d\tau, & m-1 < q < m \\ \dfrac{d^m}{dt^m} f(t), & q = m, \end{cases} \tag{2} $$

where ${}^{*}D_t^q$ is the Caputo differential operator and $\Gamma(x)$ is the gamma function.

According to the definition of fractional-order system, the fractional-order hyperchaotic Chen system can be described as

$$ \begin{cases} {}^{*}D_{t_0}^q x_1 = a(x_2 - x_1) + x_4 \\ {}^{*}Dt_0{}^q x_2 = bx_1 - x_1 x_3 + cx_2 \\ {}^{*}D_{t_0}^q x_3 = x_1 x_2 - dx_3 \\ {}^{*}D_{t_0}^q x_4 = x_2 x_3 + rx_4, \end{cases} \tag{3} $$

where q is the fractional-order. When $a = 35, b = 3, c = 12, d = 7, r = 0.5, q = 0.95$, the system is hyperchaotic.

B. ADOMIAN DECOMPOSITION METHOD

For a given fractional-order system ${}^{*}{}_{t_0}^{q}Dx(t) = f(x(t)) + g(t)$, Where $x(t) = [x_1(t), x_2(t), \cdots, x_n(t)]$ is the given function variable. $G(t) = [g_1(t), g_2(t), \cdots, g_n(t)]$ is an autonomous system, representing some constants. f is a function containing linear and nonlinear parts [40]. Therefore, any fractional chaotic system can be classified according to the above components. The form of decomposition is as follows [66]:

$$ \begin{cases} {}^{*}D_{t_0}^q x(t) = Lx(t) + Nx(t) + g(t) \\ x^{(k)}(t_0^+) = b_k, k \in [0, m-1] \\ m \in N, m-1 < q \le m, \end{cases} \tag{4} $$

where L and N represent the linear part and the nonlinear part of the system respectively, and b_k is the initial value. Applying $J_{t_0}^q$ to both sides of the Eq. (4) at the same time, the following equation can be obtained [67]:

$$ x = J_{t_0}^q Lx + J_{t_0}^q Nx + J_{t_0}^q g + \Phi, \tag{5} $$

where $\Phi = \sum_{k=1}^{m-1} b_k \frac{(t-t_0)^k}{k!}$ is the initial value condition of the system, and $J_{t_0}^q = \frac{(t-t_0)^q}{\Gamma(q+1)}$ represent the integral operator of order q. The properties of the integral operator $J_{t_0}^q$ are as follows [45]:

$$ \begin{cases} J_{t_0}^q (t-t_0)^\gamma = \dfrac{\Gamma(\gamma+1)}{\Gamma(\gamma+1+q)} (t-t_0)^{\gamma+q} \\ J_{t_0}^q C = \dfrac{C}{\Gamma(q+1)} (t-t_0)^q \\ J_{t_0}^q J_{t_0}^r x(t) = J_{t_0}^{q+r} x(t), \end{cases} \tag{6} $$

where $t \in [t_0, t_1], q \ge 0, r \ge 0, \gamma > -1$.

The nonlinear part is decomposed by Adomian decomposition algorithm as follows

$$ \begin{cases} A_j^i = \dfrac{1}{i!} [\dfrac{d^i}{d\lambda^i} N(v_j^i(\lambda))]_{\lambda=0} \\ v_j^i = \sum_{k=0} (\lambda)^k x_j^k, \end{cases} \tag{7} $$

where $i \in [0, +\infty), j \in [1, n]$.

Then the nonlinear term in Eq. (7) can be expressed as [68]

$$ Nx = \sum_{i=0}^{+\infty} A^i(x^0, x^1, \cdots, x^i). \tag{8} $$

Take Eq. (8) into Eq. (5), and the solution of the equation is as follows

$$ x = \sum_{i=0}^{+\infty} x^i = J_{t_0}^q L \sum_{i=0}^{+\infty} x^i + J_{t_0}^q L \sum_{i=0}^{+\infty} A^i(x^0, x^1, \cdots, x^i) + J_{t_0}^q g + \Phi. \tag{9} $$

The iterative relationship of the solution of Eq. (10) is as follows

$$ \begin{cases} x^0 = J_{t_0}^q g + \Phi \\ x^1 = J_{t_0}^q Lx^0 + J_{t_0}^q A^0(x^0) \\ x^2 = J_{t_0}^q Lx^1 + J_{t_0}^q A^1(x^0, x^1) \\ \cdots \\ x^i = J_{t_0}^q Lx^{i-1} + J_{t_0}^q A^{i-1}(x^0, x^1, \cdots, x^{i-1}). \\ \cdots \end{cases} \tag{10} $$

C. SOLUTION OF FRACTIONAL-ORDER HYPERCHAOTIC CHEN SYSTEM

Firstly, the fractional-order hyperchaotic Chen system Eq. (3) is decomposed into linear part L_{x_i}, nonlinear part N_{x_i} and autonomous system g_i according to the Adomian algorithm. Substituting the values of a, b, c, d and r into the equation. The decomposition results are as follows

$$ \begin{bmatrix} L_{x_1} \\ L_{x_2} \\ L_{x_3} \\ L_{x_4} \end{bmatrix} = \begin{bmatrix} 35(x_2 - x_1) + x_4 \\ 7x_1 + 12x_2 \\ -3x_3 \\ 0.5x_4 \end{bmatrix}, \begin{bmatrix} N_{x_1} \\ N_{x_2} \\ N_{x_3} \\ N_{x_4} \end{bmatrix} = \begin{bmatrix} 0 \\ -x_1 x_3 \\ x_1 x_2 \\ x_2 x_3 \end{bmatrix}, \tag{11} $$

where the autonomous system $g_i = 0$. The nonlinear terms in the system are decomposed according to Eq. (7), and the before six terms of the decomposition coefficient are

$$ \begin{cases} A_{-x_1 x_3}^0 = A_2^0 = -x_1^0 x_3^0 \\ A_{-x_1 x_3}^1 = A_2^1 = -x_1^1 x_3^0 - x_1^0 x_3^1 \\ A_{-x_1 x_3}^2 = A_2^2 = -x_1^2 x_3^0 - x_1^1 x_3^1 - x_1^0 x_3^2 \\ A_{-x_1 x_3}^3 = A_2^3 = -x_1^3 x_3^0 - x_1^2 x_3^1 - x_1^1 x_3^2 - x_1^0 x_3^3 \\ A_{-x_1 x_3}^4 = A_2^4 = -x_1^4 x_3^0 - x_1^3 x_3^1 - x_1^2 x_3^2 - x_1^1 x_3^3 \\ -x_1^0 x_3^4 \\ A_{-x_1 x_3}^5 = A_2^5 = -x_1^5 x_3^0 - x_1^4 x_3^1 - x_1^3 x_3^2 - x_1^2 x_3^3 \\ -x_1^1 x_3^4 - x_1^0 x_3^5, \end{cases} \tag{12} $$

$$\begin{cases} A^0_{x_1x_2} = A^0_3 = x^0_1x^0_2 \\ A^1_{x_1x_2} = A^1_3 = x^1_1x^0_2 + x^0_1x^1_2 \\ A^2_{x_1x_2} = A^2_3 = x^2_1x^0_2 + x^1_1x^1_2 + x^0_1x^2_2 \\ A^3_{x_1x_2} = A^3_3 = x^3_1x^0_2 + x^2_1x^1_2 + x^1_1x^2_2 + x^0_1x^3_2 \\ A^4_{x_1x_2} = A^4_3 = x^4_1x^0_2 + x^3_1x^1_2 + x^2_1x^2_2 + x^1_1x^3_2 \\ +x^0_1x^4_2 \\ A^5_{x_1x_2} = A^5_3 = x^5_1x^0_2 + x^4_1x^1_2 + x^3_1x^2_2 + x^2_1x^3_2 \\ +x^1_1x^4_2 + x^0_1x^5_2, \end{cases} \tag{13}$$

$$\begin{cases} A^0_{x_2x_3} = A^0_4 = x^0_2x^0_3 \\ A^1_{x_2x_3} = A^1_4 = x^1_2x^0_3 + x^0_2x^1_3 \\ A^2_{x_2x_3} = A^2_4 = x^2_2x^0_3 + x^1_2x^1_3 + x^0_2x^2_3 \\ A^3_{x_2x_3} = x^3_2x^0_3 + x^2_2x^1_3 + x^1_2x^2_3 + x^0_2x^3_3 \\ A^4_{x_2x_3} = A^4_4 = x^4_2x^0_3 + x^3_2x^1_3 + x^2_2x^2_3 + x^1_2x^3_3 \\ +x^0_2x^4_3 \\ A^5_{x_2x_3} = A^5_4 = x^5_2x^0_3 + x^4_2x^1_3 + x^3_2x^2_3 + x^2_2x^3_3 \\ +x^1_2x^4_3 + x^0_2x^5_3. \end{cases} \tag{14}$$

The initial conditions are $x^0_i = x_i(t_0)$. Let $c^0_i = x^0_i$. According to Eq. (3), Eq. (10) and the properties Eq.(6), x^1_i is obtained as

$$\begin{cases} x^1_1 = [35(c^0_2 - c^0_1) + c^0_4]\dfrac{(t-t_0)^q}{\Gamma(q+1)} = c^1_1\dfrac{(t-t_0)^q}{\Gamma(q+1)} \\ x^1_2 = (7c^0_1 + 12c^0_2 - c^0_1c^0_3)\dfrac{(t-t_0)^q}{\Gamma(q+1)} = c^1_2\dfrac{(t-t_0)^q}{\Gamma(q+1)} \\ x^1_3 = (c^0_1c^0_2 - 3c^0_3)\dfrac{(t-t_0)^q}{\Gamma(q+1)} = c^1_3\dfrac{(t-t_0)^q}{\Gamma(q+1)} \\ x^1_4 = (c^0_2c^0_3 + 0.5c^0_4)\dfrac{(t-t_0)^q}{\Gamma(q+1)} = c^1_4\dfrac{(t-t_0)^q}{\Gamma(q+1)}. \end{cases} \tag{15}$$

The other five coefficients can be calculated as follows

$$\begin{cases} c^2_1 = 35(c^1_2 - c^1_1) + c^1_4 \\ c^2_2 = 7c^1_1 + 12c^1_2 - c^1_1c^0_3 - c^0_1c^1_3 \\ c^2_3 = -3c^1_3 + c^1_1c^0_2 + c^0_1c^1_2 \\ c^2_4 = 0.5c^1_4 + c^1_2c^0_3 + c^0_2c^1_3, \end{cases} \tag{16}$$

$$\begin{cases} c^3_1 = 35(c^2_2 - c^2_1) + c^2_4 \\ c^3_2 = 7c^2_1 + 12c^2_2 - c^2_1c^0_3 - c^0_1c^2_3 \\ -c^1_1c^1_3\dfrac{\Gamma(2q+1)}{\Gamma^2(q+1)} \\ c^3_3 = -3c^2_3 + c^2_1c^0_2 + c^0_1c^2_2 + c^1_1c^1_2\dfrac{\Gamma(2q+1)}{\Gamma^2(q+1)} \\ c^3_4 = 0.5c^2_4 + c^2_2c^0_3 + c^0_2c^2_3 + c^1_2c^1_3\dfrac{\Gamma(2q+1)}{\Gamma^2(q+1)}, \end{cases} \tag{17}$$

$$\begin{cases} c^4_1 = 35(c^3_2 - c^3_1) + c^3_4 \\ c^4_2 = 7c^3_1 + 12c^3_2 - c^3_1c^0_3 - c^0_1c^3_3 \\ -(c^1_1c^2_3 + c^2_1c^1_3)\dfrac{\Gamma(3q+1)}{\Gamma(2q+1)\Gamma(q+1)} \\ c^4_3 = -3c^3_3 + c^3_1c^0_2 + c^0_1c^3_2 + (c^1_1c^2_2 \\ +c^2_1c^1_2)\dfrac{\Gamma(3q+1)}{\Gamma(2q+1)\Gamma(q+1)} \\ c^4_4 = 0.5c^3_4 + c^3_2c^0_3 + c^0_2c^3_3 \\ +(c^2_2c^1_3 + c^1_2c^2_3)\dfrac{\Gamma(3q+1)}{\Gamma(2q+1)\Gamma(q+1)}, \end{cases} \tag{18}$$

$$\begin{cases} c^5_1 = 35(c^4_2 - c^4_1) + c^4_4 \\ c^5_2 = 7c^4_1 + 12c^4_2 - c^4_1c^0_3 - c^2_1c^2_3\frac{\Gamma(4q+1)}{\Gamma^2(2q+1)} \\ -c^0_1c^4_3 - (c^3_1c^1_3 + c^1_1c^3_3)\dfrac{\Gamma(4q+1)}{\Gamma(3q+1)\Gamma(q+1)} \\ c^5_3 = -3c^4_3 + c^4_1c^0_2 + c^0_1c^4_2 + c^2_1c^2_2\frac{\Gamma(4q+1)}{\Gamma^2(2q+1)} \\ +(c^3_1c^1_2 + c^1_1c^3_2)\dfrac{\Gamma(4q+1)}{\Gamma(3q+1)\Gamma(q+1)} \\ c^5_4 = 0.5c^4_4 + c^4_2c^0_3 + c^0_2c^4_3 + c^2_2c^2_3\frac{\Gamma(4q+1)}{\Gamma^2(2q+1)} \\ +(c^3_2c^1_3 + c^1_2c^3_3)\dfrac{\Gamma(4q+1)}{\Gamma(3q+1)\Gamma(q+1)}, \end{cases} \tag{19}$$

$$\begin{cases} c^6_1 = 35(c^5_2 - c^5_1) + c^5_4 \\ c^6_2 = 7c^5_1 + 12c^5_2 - c^5_1c^0_3 - c^0_1c^5_3 \\ -(c^4_1c^1_3 + c^1_1c^4_3)\dfrac{\Gamma(5q+1)}{\Gamma(4q+1)\Gamma(q+1)} \\ -(c^3_1c^2_3 + c^2_1c^3_3)\dfrac{\Gamma(5q+1)}{\Gamma(3q+1)\Gamma(2q+1)} \\ c^6_3 = -3c^5_3 + c^5_1c^0_2 + c^0_1c^5_2 \\ +(c^4_1c^1_2 + c^1_1c^4_2)\dfrac{\Gamma(5q+1)}{\Gamma(4q+1)\Gamma(q+1)} \\ +(c^3_1c^2_2 + c^2_1c^3_2)\dfrac{\Gamma(5q+1)}{\Gamma(3q+1)\Gamma(2q+1)} \\ c^6_4 = 0.5c^5_4 + c^5_2c^0_3 + c^0_2c^5_3 \\ +(c^4_2c^1_3 + c^1_2c^4_3)\dfrac{\Gamma(5q+1)}{\Gamma(4q+1)\Gamma(q+1)} \\ +(c^3_2c^2_3 + c^2_2c^3_3)\dfrac{\Gamma(5q+1)}{\Gamma(3q+1)\Gamma(2q+1)}. \end{cases} \tag{20}$$

The solution of the system Eq. (4) can be expressed as follows

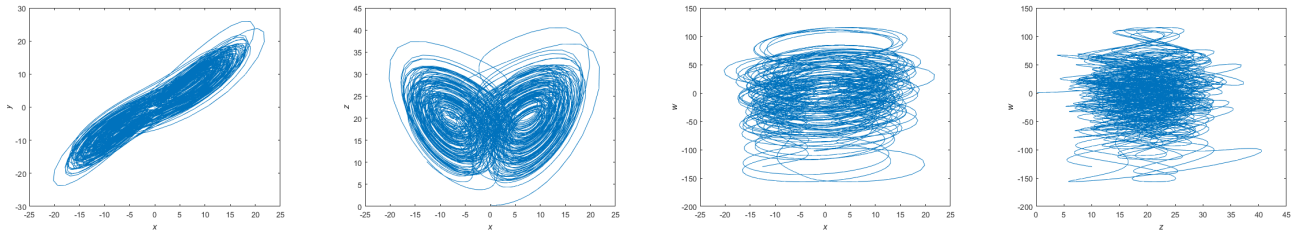
$$\begin{aligned} \widetilde{x_j}(t) = c^0_j + c^1_j\frac{(t-t_0)^q}{\Gamma(q+1)} + c^2_j\frac{(t-t_0)^{2q}}{\Gamma(2q+1)} \\ +c^3_j\frac{(t-t_0)^{3q}}{\Gamma(3q+1)} + c^4_j\frac{(t-t_0)^{4q}}{\Gamma(4q+1)} \\ +c^5_j\frac{(t-t_0)^{5q}}{\Gamma(5q+1)} + c^6_j\frac{(t-t_0)^{6q}}{\Gamma(6q+1)}, \end{aligned} \tag{21}$$

where $j = 1, 2, 3$. Fig.1 shows the attractor phase diagram of the fractional-order hyperchaotic Chen system simulated by Eq. (21).

D. DNA CODING AND COMPUTING

In biology, each DNA contains four bases: C (cytosine), T (thymine), A (adenine), and G (guanine). According to the principle of DNA base complementary pairing, A and T, C and G can be complementary pairing respectively. The coding rules are dynamically controlled by the chaotic sequences. On this basis, the addition, subtraction, XOR, and XNOR operation between DNA sequences can achieve a better encryption effect. Table.1 shows 8 ways of DNA coding [49].

Because each DNA encoding and decoding method has a corresponding DNA operation, in terms of the first encoding and decoding method, the corresponding DNA addition and subtraction operation are shown in Table.2. The XOR operations corresponding to the fourth DNA coding method are



(a) x-y (b) x-z (c) x-w (d) z-w

FIGURE 1. Attractor phase diagram of the fractional-order hyper-chaotic Chen system.(a)Attractor x-y of fractional hyperchaotic Chen system;(b)Attractor x-z of fractional hyperchaotic Chen system;(c)Attractor x-w of fractional hyperchaotic Chen system;(d)Attractor z-w of fractional hyperchaotic Chen system.

TABLE 1. Eight ways of DNA encoding and decoding.

Rules	1	2	3	4	5	6	7	8
A	00	00	01	01	10	10	11	11
T	11	11	10	10	01	01	00	00
G	01	10	00	11	00	11	01	10
C	10	01	11	00	11	00	10	01

TABLE 2. Addition and subtraction operation of first encoding method.

+	A	T	C	G	–	A	T	C	G
A	A	T	C	G	A	A	G	C	T
T	T	C	G	A	T	T	A	G	C
C	C	G	A	T	C	C	T	A	G
G	G	A	T	C	G	G	C	T	A

TABLE 3. XOR operation of fourth encoding method.

$\oplus$	A	T	C	G
A	C	G	A	T
T	G	C	T	A
C	A	T	C	G
G	T	A	G	C

shown in Table.3. The XNOR operations corresponding to the seventh DNA coding method are shown in Table.4.

TABLE 4. XNOR operation of seventh encoding method.

$\odot$	A	T	C	G
A	A	T	C	G
T	T	A	G	C
C	C	G	A	T
G	G	C	T	A

III. IMAGE COMPRESSION-ENCRYPTION AND DECRYPTION ALGORITHM

A. COMPRESSION-ENCRYPTION ALGORITHM

The flow chart of the color image compression encryption algorithm we proposed is shown in Fig.2. The color image I size used in the experiment is $M \times N$, and the specific encryption process is as follows.

Step 1: The image I is decomposed into the components of R, G, and B channels. I_1, I_2, and I_3 are three two-dimensional matrices.

Step 2: In order to process image data of any size, it is necessary to add an appropriate number of pixels with zero value to these three two-dimensional matrices. t represents the size of the segmented image block. The image size after zero fillings is given to M and N again. Each two-dimensional matrix can be divided into $(M \times N)/t^2$ image blocks.

Step 3: Setting initial values x_{00}, x_{01}, x_{02} and μ, then the sequence $\{k_i\}$, $\{k_x\}$ and $\{k_y\}$ are obtained iteratively according to Logistic mapping. The length of $\{k_i\}$ is $M \times N$, which is used for DNA operation with the original image. The length of $\{k_x\}$ and $\{k_y\}$ are M and N respectively, which are used for pixel scrambling. The formula for logistic mapping is as follows [69]

$$x_n = \mu x_{n-1}(1 - x_{n-1}), \tag{22}$$

here, when $\mu \in (3, 5699, 4]$ and $x_0 \in (0, 1)$, the system is chaotic.

The formula for calculating the initial value x_{00}, x_{01}, and x_{02} is as follows

$$\begin{cases} x_{00} = \dfrac{\sum_{i=0}^{M}\sum_{j=0}^{N} I_1(i,j) + \sum_{i=0}^{M}\sum_{j=0}^{N} I_2(i,j)}{255 \times M \times N \times 2} \\ x_{01} = \dfrac{\sum_{i=0}^{M}\sum_{j=0}^{N} I_1(i,j) + \sum_{i=0}^{M}\sum_{j=0}^{N} I_3(i,j)}{255 \times M \times N \times 2} \\ x_{02} = \dfrac{\sum_{i=0}^{M}\sum_{j=0}^{N} I_2(i,j) + \sum_{i=0}^{M}\sum_{j=0}^{N} I_3(i,j)}{255 \times M \times N \times 2}, \end{cases} \tag{23}$$

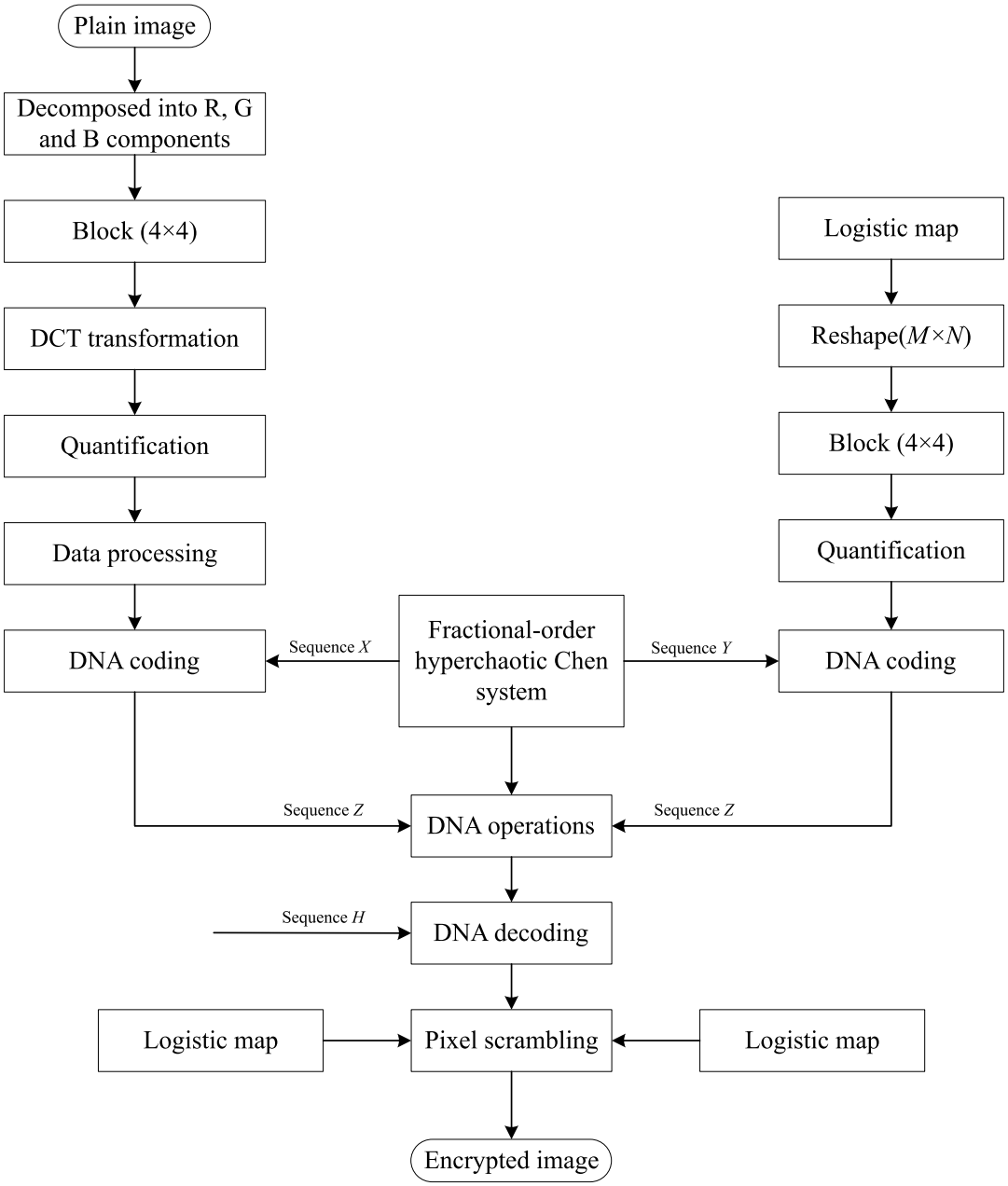


FIGURE 2. Flow chart of compression-encryption algorithm.

where x_{00} is the average gray value of I_1 and I_2, which is related to the original image and can be used as one of the keys. In the same way, x_{01} and x_{02} can be obtained.

Step 4: The sequence $\{k_i\}$ is transformed into a matrix of $M \times N$ of the same size as $I_i(i = 1, 2, 3)$, which is used for DNA operation with the original image $I_i(i = 1, 2, 3)$. In this process, the value of the matrix transformed by the chaotic sequence should be between 0 and 255, which can be obtained by Eq. (24)

$$\begin{cases} k_i = \text{mod}(\text{round}(k_i \times 10^4), 256) \\ R = \text{reshape}(k_i, N, M)'. \end{cases} \tag{24}$$

Step 5: Set the initial value of the chaotic system. These initial values are calculated according to the original image. Then, four chaotic sequences $\{X_i\}, \{Y_i\}, \{Z_i\}$ and $\{H_i\}$ can be obtained by solving the fractional-order chaotic system with the Adomian algorithm. The calculation formula of initial values is as follows

$$\begin{cases} X(0) = \dfrac{\text{sum}(\text{bitand}(I_1, 17))}{17 \times M \times N} \\ Y(0) = \dfrac{\text{sum}(\text{bitand}(I_2, 34))}{34 \times M \times N} \\ Z(0) = \dfrac{\text{sum}(\text{bitand}(I_3, 68))}{68 \times M \times N} \\ H(0) = \dfrac{\text{sum}(\text{bitand}(I_1, 136))}{136 \times M \times N}, \end{cases} \tag{25}$$

where 17 represents the binary number 00010001, bitand $(I_1, 17)$ represents the value of acquiring the first and fifth

bit-plane of I_1. The four initial values are determined by the average values of the first and fifth bit-planes of I_1, the second and sixth bit-planes of I_2, the third and seventh bit-planes of I_3 and the fourth and eighth bit-planes of I_1, respectively.

Step 6: I_1, I_2 and I_3 are divided into $t \times t$ size image blocks. In this paper, we set $t = 4$. The fourth-order DCT matrix T is used to do DCT for each sub-block, and the coefficient matrix $J_i(i = 1, 2, 3)$ in the frequency domain is obtained. The DCT formula is as follows [70]

$$J_i(u, v) = c(u)c(v) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I_i \cos \frac{(2m+1)u\pi}{2M} \times \cos \frac{(2n+1)v\pi}{2N}, \quad (26)$$

where $c(u)$ and $c(v)$ are

$$c(u) = \begin{cases} \frac{1}{\sqrt{M}}, u = 0 \\ \sqrt{\frac{2}{M}}, u \neq 0, \end{cases} \quad (27)$$

$$c(v) = \begin{cases} \frac{1}{\sqrt{N}}, v = 0 \\ \sqrt{\frac{2}{N}}, v \neq 0. \end{cases} \quad (28)$$

Step 7: The DCT transform coefficients are quantized according to the compression matrix, and different compression matrix determines different compression ratios(CR). The compression matrices used in the experiment are as follows

$$Mask1 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix},$$

$$Mask2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix},$$

$$Mask3 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix},$$

$$Mask4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix},$$

$$Mask5 = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$Mask6 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

where the compression ratio of Mask1 is 0.875, Mask2 is 0.8125, Mask3 is 0.75, Mask4 is 0.5, Mask5 is 0.4, Mask6 is 0.2. After processing, the quantized coefficient matrix $J_i'(i = 1, 2, 3)$ and $R^{'}$ are obtained.

Step 8: Processing the quantized data $J_i'(i = 1, 2, 3)$ to get $J_i''(i = 1, 2, 3)$. Since the quantized coefficients are not guaranteed to be between 0 and 255, and the DCT coefficient has positive and negative, the algorithm we proposed can adaptively adjust the quantized coefficients to meet the coding format. Taking the fractional part and sign of the coefficient matrix as one of the keys and using it in the decryption process can effectively avoid the block effect of the decrypted image caused by the direct discard. The detailed process is presented in Algorithm 1.

Algorithm 1 The Proposed Data Processing Algorithm

Input: $J_i'(i = 1, 2, 3)$: Quantized date; t: Size of the segmented image block

Output: $J_i''(i = 1, 2, 3)$: Formatted data; M_i: Fractional coefficient matrix; N_i: Sign coefficient matrix

```
Set t = 4;
Initialize matrices S and T, where S = ones(t, t), T = zeros(t, t);
Set S(1, 1) = 0;
Divide quantized date J_i' into image blocks B_i of size t×t;
L_i ← B_i × S;
find the maximum value max_num in abs(L_i);
max_Multiple ← fix(255/max_num);
find the maximum value max_num_first in abs(J_i');
max_Multiple_first ← fix(255/max_num_first);
T(1, 1) = max_Multiple_first;
for k = 2 : t do
    T(1, k) = max_Multiple;
end for
for i = 2 : t do
    for j = 1 : t do
        T(i, j) = max_Multiple;
    end for
end for
J_i' ← B_i × T;
INTJ_i ← fix(J_i');
J_i'' ← uint8(abs(INTJ_i));
M_i = J_i' − INTJ_i;
N_i = sign(INTJ_i);
```

Step 9: Perform the DNA encoding on $J_i''(i = 1, 2, 3)$ and R' respectively, which is controlled by chaotic sequence $\{X_i\}$ and $\{Y_i\}$. Then the results of DNA coding are calculated by DNA operations. There are four kinds of operations, which are controlled by chaotic sequence $\{Z_i\}$. Finally, the process of DNA decoding is controlled by chaotic sequence $\{H_i\}$. Among the four chaotic sequences, $\{X_i\}$, $\{Y_i\}$ and $\{H_i\}$ are between 1 and 8 and $\{Z_i\}$ is between 0 and 3, which is transformed by the following formula

$$\begin{cases} X_i = \text{mod}(\text{round}(X_i \times 10^4), 8) + 1 \\ Y_i = \text{mod}(\text{round}(Y_i \times 10^4), 8) + 1 \\ Z_i = \text{mod}(\text{round}(Z_i \times 10^4), 4) + 1 \\ H_i = \text{mod}(\text{round}(H_i \times 10^4), 8) + 1, \end{cases} \quad (29)$$

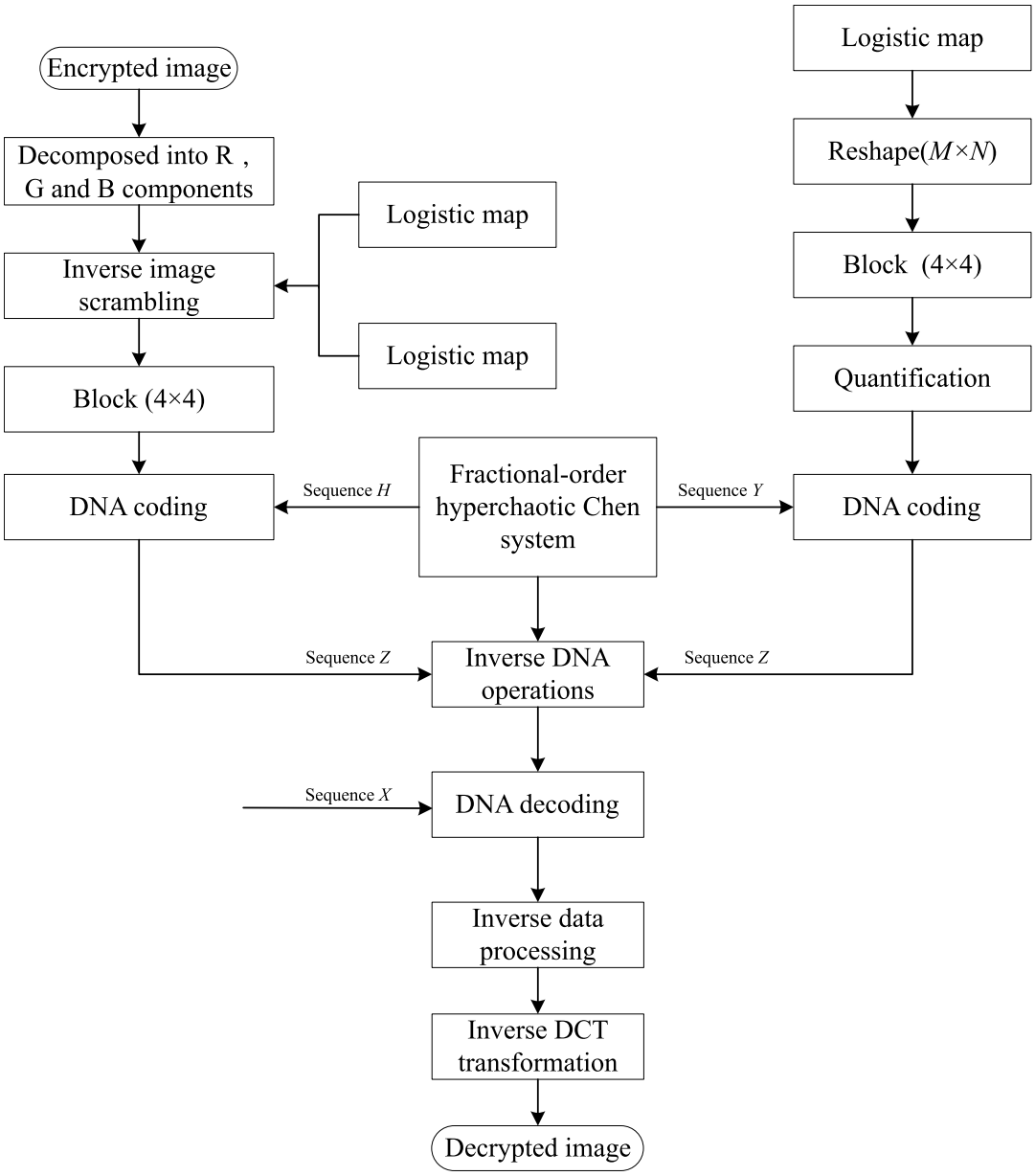


FIGURE 3. Flow chart of decryption algorithm.

where $\{X_i\}$, $\{Y_i\}$ and $\{H_i\}$ sequence values correspond to 8 ways of DNA encoding and decoding, and $\{Z_i\}$ sequence values correspond to 4 ways of DNA operations. Except for the first sub block, the DNA calculation of the result of the current sub-block and the last sub-block is performed again to obtain a better diffusion effect, which controlled by the sequence $\{Z_i\}$.

Step 10: Pixel scrambling. The sequences $\{k_x\}$ and $\{k_y\}$ are arranged in descending order. The sequence $\{U_x\}$ and $\{U_y\}$ composed of the index values of the sorted elements in the original sequence are obtained. Taking the $\{U_x\}$, $\{U_y\}$ sequence values and their corresponding indexes as row and column exchange coordinates, row and column permutations are performed on the matrices of the three channels after DNA decoding. The detailed process is presented in Algorithm 2.

Step 11: Combining three two-dimensional matrices to get encrypted image.

B. DECRYPTION ALGORITHM

The decryption process is the reverse operation of the encrypted image, and the decrypted image can only be obtained by using the same key as the encrypted image. The flow chart of the decryption algorithm we proposed is shown in Fig.3.

There are several points to pay attention to when decrypting. Firstly, in the case of inverse pixel scrambling, the order of row and column permutation is opposite to that

of encryption. Secondly, in encryption, the DNA decoding method of the image is determined by $\{H_i\}$. So in decryption, the DNA encoding method of the ciphertext image is also determined by $\{H_i\}$. Thirdly, when performing the inverse DNA operation, if the addition operation is used in encryption, the subtraction operation should be used in decryption, and vice versa. Finally, the zero pixels added during encryption should be removed during decryption.

Algorithm 2 The Proposed Pixel Scrambling Algorithm

Input: k_x: Chaotic sequence with length M; k_y: Chaotic sequence with length N; $Q_n(n = 1, 2, 3)$: Matrix of three channels after DNA decoding;

Output: $Q'_n(n = 1, 2, 3)$: Matrix of three channels after pixel scrambling;

1: $[\sim, U_x] = \text{sort}(k_x, \text{descend})$;
2: $[\sim, U_y] = \text{sort}(k_y, \text{descend})$;
3: **for** $i = 1 : M$ **do**
4: $\quad temp = Q_n(i, :)$;
5: $\quad Q_n(i, :) = Q_n(U_x(i), :)$;
6: $\quad Q_n(U_x(i), :) = temp$;
7: **end for**
8: **for** $i = 1 : N$ **do**
9: $\quad temp = Q_n(:, i)$;
10: $\quad Q_n(:, i) = Q_n(:, U_y(i))$;
11: $\quad Q_n(:, U_y(i)) = temp$;
12: **end for**
13: Denote Q_n as Q'_n.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

This part introduces the detailed experimental results and performance analysis of our compression-encryption algorithm. All the experimental results are run on MATLAB 2015b in a personal computer with Intel(R) Core(TM) i5-5200, CPU 2.20GHZ and memory 6.00GB, and the operating system is Microsoft Windows 10. When the parameters of chaotic system set in the experiment are: $a = 35, b = 3, c = 12, d = 7, r = 0.5, q = 0.95$, the system is hyperchaotic. Taking Lena (256×256) as the input image, the four initial values of the chaotic system are calculated by the input image, which are as follows: $X(0) = 0.4979$, $Y(0) = 0.4276$, $Z(0) = 0.7086$ and $H(0) = 0.7807$. We select six color images (256×256) as experimental images, which are ''Lena'', ''peppers'', ''baboon'', ''airplane'', ''house'' and ''lake''. The encrypted and decrypted images are shown in Fig.4. It can be seen that the algorithm can encrypt images effectively.

A. COMPRESSION PERFORMANCE ANALYSIS

Six compression matrices are used in the experiment, and the compression ratios are 0.875, 0.8125, 0.75, 0.5, 0.4 and 0.2 respectively. Peak signal-to-noise ratio (PSNR) is used to measure the quality of the decompressed image. Generally, the larger PSNR is, the smaller the distortion of the decompressed image is. The calculation formula of PSNR index is

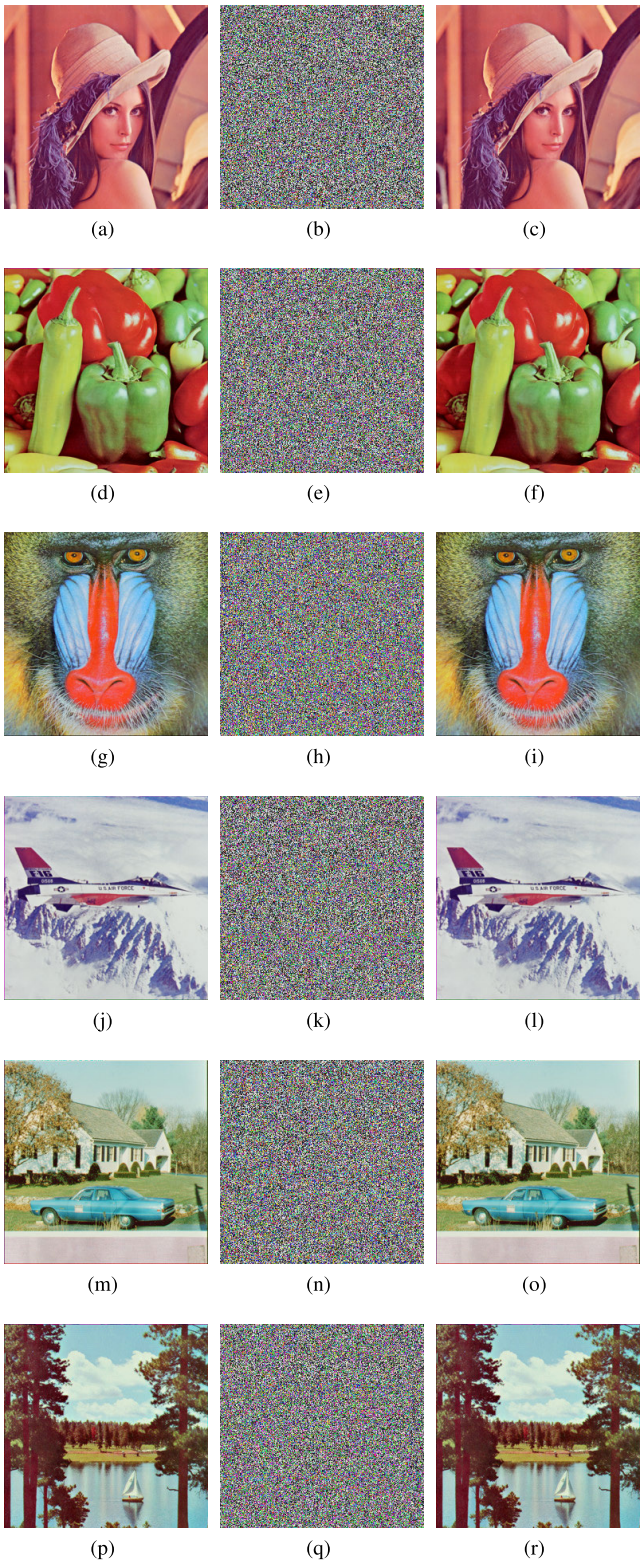


FIGURE 4. Experimental simulation results. (a) original Lena image; (b) encrypted Lena image; (c) decrypted Lena image; (d) original peppers image; (e) encrypted peppers image; (f) decrypted peppers image; (g) original baboon image; (h) encrypted baboon image; (i) decrypted baboon image; (j) original airplane image; (k) encrypted airplane image; (l) decrypted airplane image; (m) original house image; (n) encrypted house image; (o) decrypted house image; (p) original lake image; (q) encrypted lake image; (r) decrypted lake image.

as follows [49]

$$\begin{cases} \text{MSE} = \dfrac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} (P'(i,j) - P(i,j))^2 \\ \text{PSNR} = 10\log_{10}(\dfrac{255^2}{\text{MSE}}) \end{cases}, \quad (30)$$

where MSE represents mean square error, which is used to describe the deviation between the estimated value and the real value, $P'(i,j)$ represents the encrypted image, and $P(i,j)$ represents the image before encryption. According to Eq. (30), PSNR of three channels can be calculated.

The experimental results of encryption and decryption under different compression ratios are shown in Fig.5. We combined the proposed encryption algorithm with CS, and compared with our algorithm. The discrete wavelet transform (DWT) is used for sparse representation of the image, the measurement matrix is obtained from partial Hadamar matrix, and the orthogonal matching pursuit (OMP) algorithm is used to reconstruct the image. The change of encrypted image data size with compression ratio is shown in Table.5, and the corresponding PSNR indexes are shown in Table.6. It can be seen from Table.5 and Table.6 that with the decrease of the compression rate, the amount of data and PSNR index of encrypted image decrease gradually. Although the compression algorithm based on CS can obtain smaller encrypted image, the PSNR index of reconstructed image is not ideal. As can be seen from Table.6, the PSNR index of the decompressed image is still greater than 30dB when CR $= 0.2$ in our algorithm, which shows that it can reconstruct the original image well.

B. HISTOGRAM ANALYSIS

Gray histogram reflects the statistical feature of gray value and frequency in the image. Taking the components of R channel of each color picture as an example, the histogram before encryption and after decryption is shown in Fig.6, where (a), (b), (c), (g), (h) and (i) represent the histogram of original image, (d), (e), (f), (j), (k) and (l) represent the histogram of decrypted image. In general, we hope that the histogram of the plaintext image and the encrypted image have a great change, and the histogram of the encrypted image of different plaintext images has a similar structure. The theoretical analysis of the variance of the histogram [71] can further illustrate the anti-statistical attack performance of the algorithm. The calculation formula of histogram variance is as follows [38]

$$\text{var}(C) = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} (c_i - c_j), \quad (31)$$

where n is the greyness level, $C = \{c_0, c_1, \cdots, c_{255}\}$ denotes the vector of the histogram value and c_i and c_j are the quantity of the gray value. Taking the R channel of each image as an example, the calculation results of different algorithms are shown in Table.7. From the experimental results, our proposed encryption algorithm reflects this requirement well.

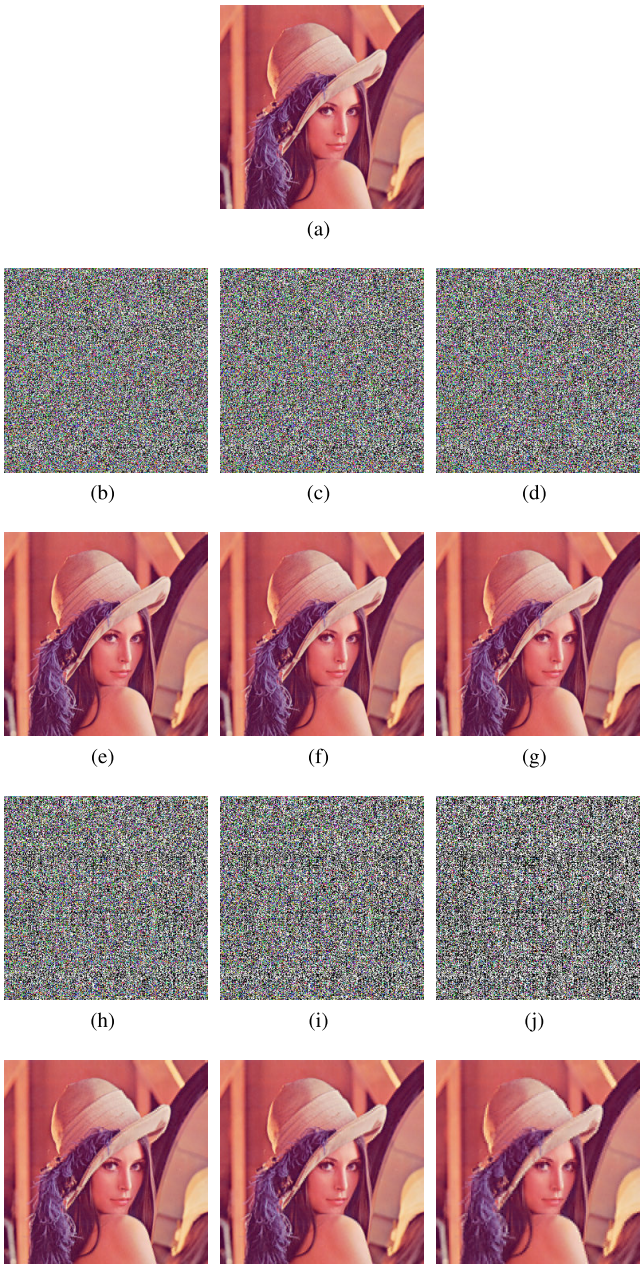


FIGURE 5. Decrypted image with different compression ratio. (a) original Lena image; (b) encrypted Lena of CR = 0.875; (c) encrypted Lena of CR = 0.8125; (d) encrypted Lena of CR = 0.75; (e) decrypted Lena of CR = 0.875; (f) decrypted Lena of CR = 0.8125; (g) decrypted Lena of CR = 0.75; (h) encrypted Lena of CR = 0.5; (i) encrypted Lena of CR = 0.4; (j) encrypted Lena of CR = 0.2; (k) decrypted Lena of CR = 0.5; (l) decrypted Lena of CR = 0.4; (m) decrypted Lena of CR = 0.2.

It shows that our encryption algorithm can resist the statistical attack commendably.

C. CORRELATION ANALYSIS OF ADJACENT PIXELS

The correlation of adjacent pixels reflects the correlation degree of pixel values in adjacent positions of the image, which is one of the important statistical characteristics of

TABLE 5. Encrypted image data size(KB) with different CR.

CR(%)	100	87.5	81.25	75	50	40	20
Ours	192	181	176	167	129	109	73.1
Compression based on CS	192	168	156	144	96.3	78.3	54.2

TABLE 6. PSNR(dB) index under different CR.

CR(%)	Ours			Compression based on CS		
	R	G	B	R	G	B
87.5	48.5877	48.5877	48.5877	29.6754	29.1375	29.7969
81.25	45.6176	45.6176	45.6176	28.9972	28.3609	29.1564
75	41.5266	41.5266	41.5266	28.3067	27.6168	28.4498
50	35.5644	35.5644	35.5644	24.6024	23.4531	24.7563
40	34.6794	34.6794	34.6794	22.5685	21.0029	22.9113
20	30.1952	30.1952	30.1952	4.2604	9.5979	11.4226

TABLE 7. Comparison variance by different algorithms(R channel).

Variance	Plain image	Ref.[5]	Ref.[12]	Ref.[33]	Ours
Lena	130029.1	519.2	387.3	1229.5	542.3
Pappers	114211.9	489.5	536.8	1017.9	480.2
Baboon	57779.4	541.1	560.6	1180.9	534.5
Airplane	334365.8	559.3	504.2	1040.0	475.5
House	97792.1	466.7	501.9	1053.9	487.2
Lake	108303.4	535.9	512.3	1166.8	545.6

the image. The smaller the correlation between the adjacent pixels of the encrypted image, the better the performance of the designed encryption system. Then 10000 pixels are selected from plaintext image and encrypted image to test the pixel correlation. The correlation r_{xy} of two adjacent pixels is calculated as follows [50]

$$\begin{cases} r_{xy} = \dfrac{cov(x,y)}{\sqrt{D(x)D(y)}} = \dfrac{E\{[x-E(x)][y-E(y)]\}}{\sqrt{D(x)D(y)}} \\ E(x) = \dfrac{1}{M}\sum_{i=1}^{M} x_i \\ D(x) = \dfrac{1}{M}\sum_{i=1}^{M}[x_i - E(x)]^2, \end{cases} \tag{32}$$

where $r_{xy} \in [0, 1]$, and the larger r_{xy} is, the higher the correlation between adjacent pixels is.

Taking the R components of Lena as an example, the pixel distribution in the horizontal, vertical and diagonal directions before and after encryption is shown in Fig.7. Observing

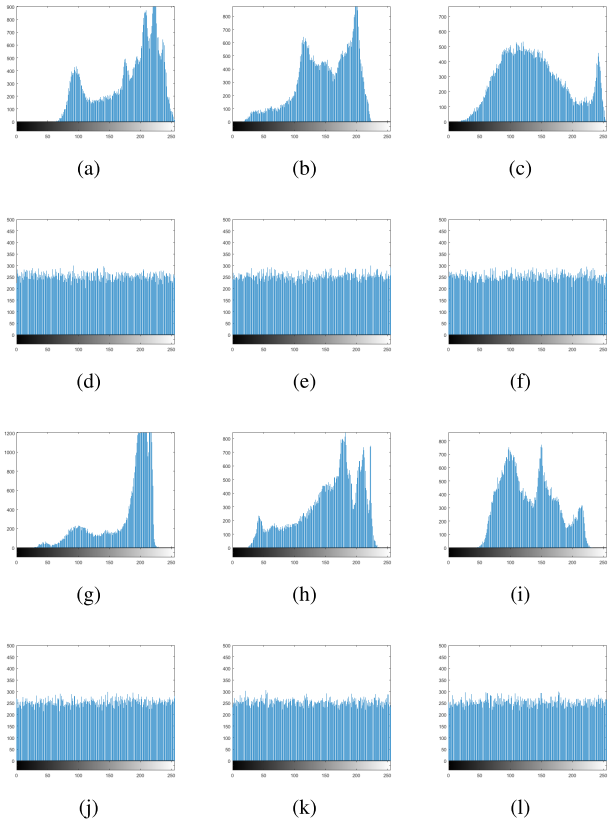


FIGURE 6. Histogram before and after encryption of R channel. (a) original Lena of R channel; (b) original peppers of R channel; (c) original baboon of R channel; (d) encrypted Lena of R channel; (e) encrypted peppers of R channel; (f) encrypted baboon of R channel; (g) original airplane of R channel; (h) original house of R channel; (i) original lake of R channel; (j) encrypted airplane of R channel; (k) encrypted house of R channel; (l) encrypted lake of R channel.

Fig.7, after encryption, the pixel distribution of the image is particularly chaotic, and there is no rule to follow. Table.8 lists the adjacent pixel correlations of different cipher images with our proposed encryption algorithm. Table.9 lists the adjacent pixel correlation of encrypted Lena image with different encryption algorithms. From the data listed in Table.8, it can be seen that the correlation of adjacent pixels in all directions of the encrypted image is close to 0, which shows that our proposed encryption algorithm can effectively reduce the correlation of adjacent pixels and improve the security of the encryption system. Moreover, Table.9 shows the comparison results of our encryption algorithm with those in references [5], [12] and [33]. It can be seen that the pixel correlation of our algorithm is similar to that of other algorithms, and the correlation in some directions is even smaller, which indicates that our algorithm has excellent encryption performance.

D. KEY SPACE

The key space is used to measure the resistance of the encryption system to the brute-force attack. For an encryption system, the key space needs to be greater than 2^{100} to

TABLE 8. Pixel correlation coefficients of plain and cipher images.

Image	Direction	Pixel correlation coefficients					
		Plain images			Cipher images		
		R	G	B	R	G	B
Lena	Horizontal	0.9592	0.9438	0.9265	0.0071	-0.0012	-0.0015
	Vertical	0.9788	0.9697	0.9529	0.0089	-0.0018	0.0041
	Diagonal	0.9351	0.9169	0.8969	-0.0006	-0.0043	-0.0041
Peppers	Horizontal	0.9633	0.9706	0.9566	0.0049	0.0027	-0.0019
	Vertical	0.9678	0.9765	0.9641	0.0031	-0.0100	0.0072
	Diagonal	0.9372	0.9495	0.9295	0.0037	0.0058	-0.0027
Baboon	Horizontal	0.9484	0.8722	0.9217	0.0055	0.0091	-0.0035
	Vertical	0.0092	0.8386	0.9152	-0.0018	0.0061	0.0047
	Diagonal	0.9051	0.7903	0.8761	-0.0015	0.0035	0.00004
Airplane	Horizontal	0.9370	0.9270	0.9464	0.0013	0.0047	0.0085
	Vertical	0.9244	0.9330	0.9069	0.0015	0.0189	0.0014
	Diagonal	0.8762	0.8803	0.8804	-0.0036	0.0038	0.0193
House	Horizontal	0.9292	0.9073	0.9542	-0.0014	0.0195	-0.0079
	Vertical	0.9252	0.9183	0.9489	-0.0041	0.0211	-0.0015
	Diagonal	0.8809	0.8483	0.9147	0.0002	-0.0008	0.0023
Lake	Horizontal	0.9567	0.9552	0.9616	-0.0017	-0.0039	0.0029
	Vertical	0.9563	0.9551	0.9675	0.0132	-0.0044	0.0027
	Diagonal	0.9297	0.9265	0.9404	0.0038	-0.0019	0.0057

TABLE 9. Pixel correlation comparison for Lena (256 × 256).

Channels	Direction	Plain image	Ref.[5]	Ref.[12]	Ref.[33]	Ours
R	Horizontal	0.9592	-0.0145	-0.0151	0.0035	0.0071
	Vertical	0.9788	-0.0066	-0.0051	-0.0014	0.0089
	Diagonal	0.9351	0.0013	-0.0026	0.0415	-0.0006
G	Horizontal	0.9438	0.0022	0.0109	0.0029	-0.0012
	Vertical	0.9697	0.0106	0.0007	0.0040	-0.0018
	Diagonal	0.9169	-0.0108	-0.0019	0.0031	-0.0043
B	Horizontal	0.9265	-0.0098	-0.0179	0.0029	-0.0015
	Vertical	0.9529	-0.0094	0.0001	0.0040	0.0041
	Diagonal	0.8969	0.0174	-0.0069	0.0031	-0.0041

ensure security. In our encryption algorithm, the key space consists of block size t, the parameters and initial values of logistic chaotic system μ, x_{00}, x_{01} and x_{02}, the initial values of fractional-order hyperchaotic Chen system $X(0)$,

TABLE 10. Sensitivity of different keys.

Keys	μ	x_{00}	x_{01}	x_{02}	X(0)	Y(0)	Z(0)	H(0)	p
Sensitivity	10^{-16}	10^{-15}	10^{-16}	10^{-16}	10^{-14}	10^{-16}	10^{-14}	10^{-14}	10^{-16}

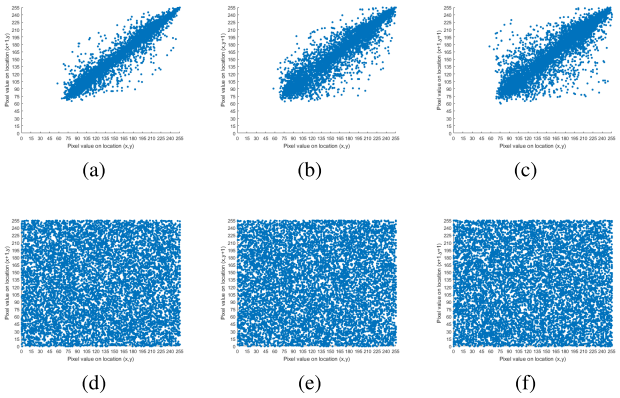


FIGURE 7. Distribution of adjacent pixels in plain and cipher image. (a) Horizontal direction on the R components of Lena; (b) Vertical direction on the R components of Lena; (c) Diagonal direction on the R components of Lena; (d) Horizontal direction on the R components of Lena; (e) Vertical direction on the R components of Lena; (f) Diagonal direction on the R components of Lena.

$Y(0)$, $Z(0)$ and $H(0)$, fractional-order q, the number of 0 pixels added. The key capacity of this encryption algorithm is $10^{137} > 2^{455}$. In addition, you can change the parameter μ of three logistic maps to make them different to further enhance the key space. Therefore, our algorithm can effectively resist brute force attacks.

E. KEY SENSITIVITY ANALYSIS

The basic requirement for the encryption system is to have a high sensitivity to the key, even if the input key changes to a very small extent, it can't get the correct decryption image. Taking the Lena image as an example, we test the key sensitivity of μ, x_{00}, x_{01}, x_{02}, $X(0)$, $Y(0)$, $Z(0)$, $H(0)$, and p. The corresponding key sensitivity is shown in Table.10. We test whether the correct decryption image can be obtained by only changing a single key slightly. (By default, the symbol matrix in step 8 of the encryption process is not known.) The test results are shown in Fig.8, which can be seen that even if the key is changed to a very small extent, the correct decryption image can't be obtained. So our encryption system is very sensitive to the key.

F. INFORMATION ENTROPY

In the image encryption algorithm, information entropy is usually used as an index to evaluate the randomness of the image. The formula of information entropy is as follows

$$H(x) = -\sum_{i=0}^{2^N-1} p(x_i)\log_2 p(x_i), \tag{33}$$

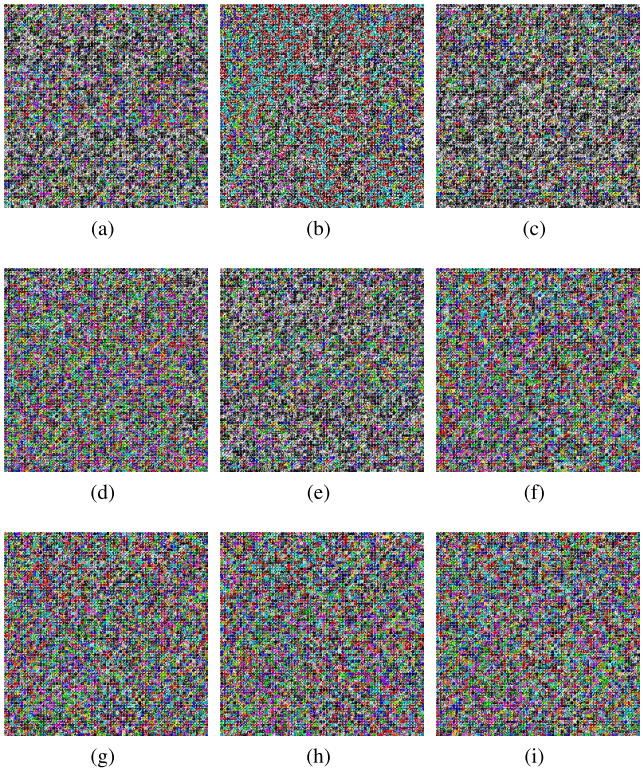


FIGURE 8. Decryption image with key changed slightly. (a) Lena with $\mu + 10^{-16}$; (b) Lena with $x_0 + 10^{-15}$; (c) Lena with $X(0) + 10^{-14}$; (d) Lena with $Y(0) + 10^{-16}$; (e) Lena with $Z(0) + 10^{-14}$; (f) Lena with $H(0) + 10^{-14}$; (g) Lena with $x_1 + 10^{-16}$; (h) Lena with $x_2 + 10^{-16}$; (i) Lena with $p + 10^{-16}$.

where $p(x_i)$ represents the proportion of pixel in the image whose pixel value is x_i, and 2^N represents the gray level of the image. When the gray level is $256 = 2^8$, the theoretical value of information entropy $H_{max} = \log_2 256 = 8$.

Table.11 shows the information entropy comparison of different plain images and cipher images. It can be seen that the information entropy of the encrypted image is closer to the theoretical value than the image before encryption, which shows that our encryption algorithm can better resist the information entropy attack. The comparison results of information entropy with other encryption algorithms are shown in Table.12. From the experimental data in Table.12, we can see that the average information entropy of our algorithm is better than other algorithms.

G. DIFFERENTIAL ATTACK

In order to detect whether the encryption system has the ability to resist differential attack, number of pixels change

TABLE 11. Information entropy comparison of different images.

Image	Plain image			Cipher image		
	R	G	B	R	G	B
Lena	7.2417	7.5777	6.9171	7.9970	7.9972	7.9967
Peppers	7.3009	7.5570	7.0929	7.9974	7.9972	7.9971
Baboon	7.6058	7.3581	7.6665	7.9971	7.9977	7.9970
Airplane	6.7254	6.8253	6.2078	7.9974	7.9972	7.9980
House	7.4052	7.2317	7.4280	7.9974	7.9972	7.9970
Lake	7.2587	7.6143	7.1892	7.9970	7.9967	7.9969

TABLE 12. Comparison of information entropy with other algorithms for Lena.

Image	Information entropy			
	R	G	B	Average
Lena	7.2417	7.5777	6.9171	7.2458
Ref.[5]	7.9971	7.9970	7.9966	7.9969
Ref.[33]	7.9932	7.9941	7.9941	7.9938
Ours	7.9970	7.9972	7.9967	7.9970

rate (NPCR) and unified average changing intensity (UACI) are usually used to measure. The calculation formulas of these two indicators are as follows [36]

$$\mathrm{NPCR}_n = \frac{\sum_{i,j} D_n(i,j)}{M \times N} \times 100\%, \tag{34}$$

$$\mathrm{UACI}_n = \frac{1}{MN} \times [\sum_{i,j} \frac{|c'_n(i,j) - c_n(i,j)|}{255}] \times 100\%, \tag{35}$$

$$\mathrm{D}_n(i,j) = \begin{cases} 1, c'_n(i,j) - c_n(i,j) \neq 0 \\ 0, c'_n(i,j) - c_n(i,j) = 0, \end{cases} \tag{36}$$

where n is an integer and $n \in [1, 3]$, $n = 1$, 2 and 3 represent the NPCR and UACI indexes of R, G and B channels respectively. C'_n represents the encrypted image after the original image C_n changes one pixel value.

We just change the value of a single pixel in the plain image to observe the NPCR and UACI indexes of the two encrypted images. The results of the two indexes are listed in Table.13, which can be seen that the NPCR and UACI indexes of the encrypted image are very close to the ideal values, so it shows that the encryption algorithm can well resist the differential attack. Table.14 shows the comparison results between our algorithm and other algorithms based on the Lena image. From the experimental data in Table.14, it can be seen that the NPCR and UACI indexes of the comparison algorithm are poor when only changing the single pixel value, which can not resist the differential attack effectively. In contrast, our algorithm greatly improves the performance of resisting differential attack.

TABLE 13. NPCR and UACI of different images.

Images	NPCR(%)			UACI(%)		
	R	G	B	R	G	B
Lena	99.6033	99.6292	99.5850	33.4740	33.4457	33.5339
Peppers	99.6277	99.6414	99.6216	33.5270	33.6326	33.4782
Baboon	99.5956	99.6109	99.6063	33.2395	33.3200	33.3482
Airplane	99.6140	99.6063	99.6140	33.4113	33.5652	33.4265
House	99.5865	99.6033	99.5956	33.5184	33.5291	33.5091
Lake	99.5941	99.6109	99.5834	33.4398	33.5558	33.3875

TABLE 14. Comparison of NPCR and UACI for the Lena by different algorithms.

Images	NPCR(%)			UACI(%)		
	R	G	B	R	G	B
Ref.[12]	0.00153	0.00153	0.00153	0.00097	0.00071	0.00008
Ref.[33]	0.00458	0.00458	0.00458	0.00223	0.00201	0.00201
Ours	99.6033	99.6292	99.5850	33.4740	33.4457	33.5339

H. RESISTING KNOWN-PLAINTEXT AND CHOSEN-PLAINTEXT ATTACKS ABILITY

From the previous analysis, because the parameters used for encryption in these algorithms are not sensitive to plaintext [10], [28], [33], they have been cracked by known plaintext attack and selective plaintext attack, which are often used to evaluate the security of the algorithm [73]. In our algorithm, we improved the performance of resisting these two attacks through corresponding operations. Firstly, our encryption algorithm is based on fractional-order hyperchaotic Chen system and logistic map, and its initial values are completely determined by plaintext. Secondly, our encryption algorithm is highly sensitive to the keys, and the chaotic sequences $\{X_i\}$, $\{Y_i\}$, $\{Z_i\}$ and $\{H_i\}$ can dynamically control each encryption process. This means that when the plaintext changes, the keys will change, which will lead to the change of chaotic sequence, so the encryption result will also alter. Therefore, our algorithm is highly dependent on the original image and can resist these two attacks.

I. ANTI-NOISE ABILITY

In the process of transmission, the encrypted image will inevitably be disturbed by noise. The typical noise types are Gaussian noise (GN) and salt pepper noise (SPN). The encrypted image and the corresponding decrypted image

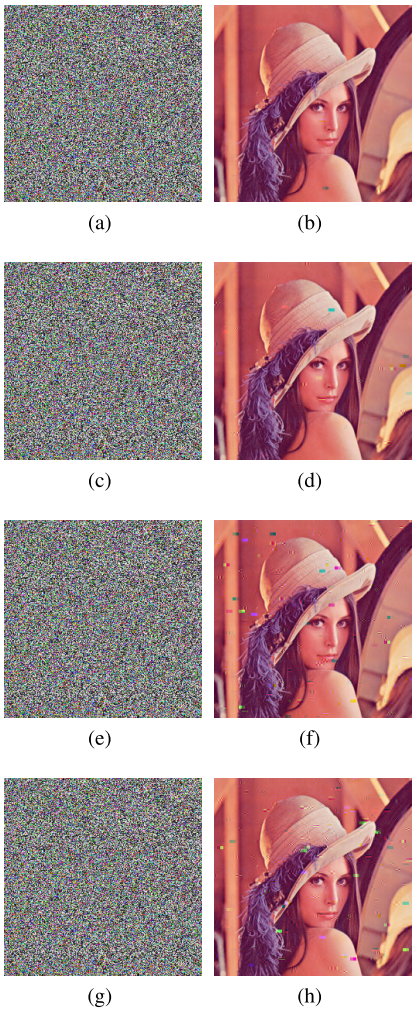


FIGURE 9. Experimental results of anti-noise performance. (a) encrypted image with Gaussian noise variance of 0.15; (b)decrypted image with Gaussian noise variance of 0.15; (c) encrypted image with Gaussian noise variance of 0.2; (d) decrypted image with Gaussian noise variance of 0.2; (e) encrypted image with salt and pepper noise density of 0.002; (f) decrypted image with salt and pepper noise density of 0.002; (g) encrypted image with salt and pepper noise density of 0.005; (h) decrypted image with salt and pepper noise density of 0.005.

under different noise densities are shown in Fig.9. The variance of Gaussian noise is 0.15 and 0.20, respectively. The density of salt and pepper noise was 0.002 and 0.005, respectively. It can be seen from Fig.9 that most of the original image information can be recovered after decrypting the cipher image with noise, so our algorithm has good anti-noise performance.

J. ANTI-CROPPING ABILITY

When transmitting an encrypted image, it is easy to be attacked by occlusion. Therefore, we test the decrypted images with different degrees of data loss, which are $\frac{1}{8}$ of the occluded encrypted images, $\frac{1}{16}$ of the occluded encrypted images and $\frac{1}{32}$ of the occluded encrypted images to detect the anti-cropping attack performance of our algorithm. The experimental results are shown in Fig.10. It can be seen

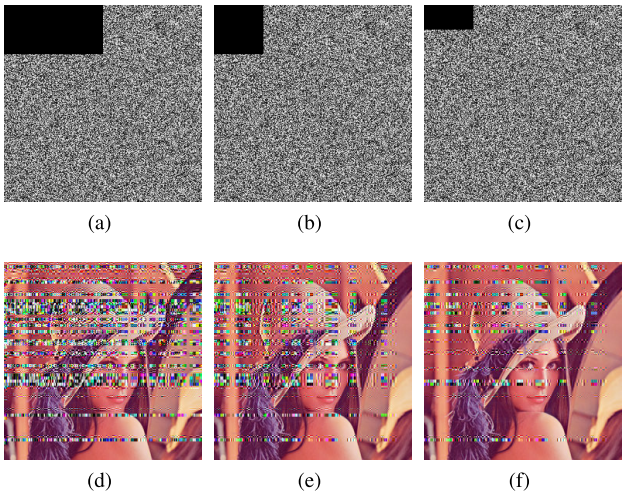


FIGURE 10. Experimental results of anti-noise performance. (a) 1/8 of the encrypted image is cropped; (b) 1/16 of the encrypted image is cropped; (c) 1/32 of the encrypted image is cropped; (d) decrypted image of 1/8 data cropped; (e) decrypted image of 1/16 data cropped; (f) decrypted image of 1/32 data cropped.

from Fig.10 that although a part of the encrypted image has been cut off, our algorithm can still recover the original image well.There are some flaws in the decrypted image, but most of the original image information can be obtained from it, which shows that our proposed algorithm has good anti-cropping attack ability.

V. CONCLUSION

In this paper, we proposed a novel color image compression-encryption algorithm based on the fractional-order hyperchaotic system combined with DCT and DNA coding. In the compression stage, the two-dimensional DCT transforms the image to the frequency domain for processing, and the quantization operation is used to reduces the amount of image data. In the encryption stage, chaotic sequences are generated by fractional-order hyperchaotic Chen system, which are used to control DNA coding, decoding and DNA perations. Moreover, the logistic map is used to scramble the pixel position, which further improves the security of the algorithm. Experimental results verify that our algorithm can effectively compress image data and resist various attacks. However, although the high-dimensional fractional-order hyperchaotic system is used in the encryption algorithm to ensure that it has enough key space, the process of pixel scrambling is only controlled by the one-dimensional logistic map, and its parameter space is very limited, which may affect the key space. Secondly, we only use statistical tests to analyze the security of encryption algorithm, which are necessary but not sufficient. These shortcomings will be improved in future research.

REFERENCES

[1] L. Liu, Y. Zhang, and X. Wang, "A novel method for constructing the S-Box based on spatiotemporal chaotic dynamics," *Appl. Sci.*, vol. 8, no. 12, p. 2650, Dec. 2018.

[2] Y. Zhou, L. Bao, and C. L. P. Chen, "A new 1D chaotic system for image encryption," *Signal Process.*, vol. 97, pp. 172–182, Apr. 2014.

[3] S. Toughi, M. H. Fathi, and Y. A. Sekhavat, "An image encryption scheme based on elliptic curve pseudo random and advanced encryption system," *Signal Process.*, vol. 141, pp. 217–227, Dec. 2017.

[4] E. N. Lorenz, "The mechanics of vacillation," *J. Atmos. Sci.*, vol. 20, no. 5, pp. 448–465, Sep. 1963.

[5] S. Rohith, K. N. H. Bhat, and A. N. Sharma, "Image encryption and decryption using chaotic key sequence generated by sequence of logistic map and sequence of states of linear feedback shift register," in *Proc. Int. Conf. Adv. Electron. Comput. Commun.*, Bengaluru, India, Oct. 2014, pp. 1–6.

[6] J. Sun, X. Zhao, J. Fang, and Y. Wang, "Autonomous memristor chaotic systems of infinite chaotic attractors and circuitry realization," *Nonlinear Dyn.*, vol. 94, no. 4, pp. 2879–2887, Aug. 2018.

[7] J. Sun, Y. Wu, G. Cui, and Y. Wang, "Finite-time real combination synchronization of three complex-variable chaotic systems with unknown parameters via sliding mode control," *Nonlinear Dyn.*, vol. 88, no. 3, pp. 1677–1690, Feb. 2017.

[8] S. Hraoui, F. Gmira, M. F. Abbou, A. J. Oulidi, and A. Jarjar, "A new cryptosystem of color image using a dynamic-chaos hill cipher algorithm," *Procedia Comput. Sci.*, vol. 148, pp. 399–408, Jan. 2019.

[9] Z. Hua and Y. Zhou, "Image encryption using 2D logistic-adjusted-sine map," *Inf. Sci.*, vol. 339, pp. 237–253, Apr. 2016.

[10] C. Pak and L. Huang, "A new color image encryption using combination of the 1D chaotic map," *Signal Process.*, vol. 138, pp. 129–137, Sep. 2017.

[11] Y. Zhang, "The unified image encryption algorithm based on chaos and cubic S-Box," *Inf. Sci.*, vol. 450, pp. 361–377, Jun. 2018.

[12] M. B. Hossain, M. T. Rahman, A. B. M. S. Rahman, and S. Islam, "A new approach of image encryption using 3D chaotic map to enhance security of multimedia component," in *Proc. Int. Conf. Informat., Electron. Vis. (ICIEV)*, May 2014, pp. 1–6.

[13] Y.-Q. Zhang, Y. He, P. Li, and X.-Y. Wang, "A new color image encryption scheme based on 2DNLCML system and genetic operations," *Opt. Lasers Eng.*, vol. 128, May 2020, Art. no. 106040.

[14] Y. He, Y.-Q. Zhang, and X.-Y. Wang, "A new image encryption algorithm based on two-dimensional spatiotemporal chaotic system," *Neural Comput. Appl.*, vol. 32, no. 1, pp. 247–260, Jan. 2020.

[15] Y.-Q. Zhang and X.-Y. Wang, "A new image encryption algorithm based on non-adjacent coupled map lattices," *Appl. Soft Comput.*, vol. 26, pp. 10–20, Jan. 2015.

[16] Y. Zhang, Y. He, and X. Wang, "Spatiotemporal chaos in mixed linear–nonlinear two-dimensional coupled logistic map lattice," *Phys. A, Stat. Mech. Appl.*, vol. 490, pp. 148–160, Jan. 2018.

[17] G. Ye and J. Zhou, "A block chaotic image encryption scheme based on self-adaptive modelling," *Appl. Soft Comput.*, vol. 22, pp. 351–357, Sep. 2014.

[18] Y.-G. Yang, L. Zou, Y.-H. Zhou, and W.-M. Shi, "Visually meaningful encryption for color images by using qi hyper-chaotic system and singular value decomposition in YCbCr color space," *Optik*, vol. 213, Jul. 2020, Art. no. 164422, doi: 10.1016/j.ijleo.2020.164422.

[19] A. Broumandnia, "The 3D modular chaotic map to digital color image encryption," *Future Gener. Comput. Syst.*, vol. 99, pp. 489–499, Oct. 2019.

[20] X. Chai, X. Fu, Z. Gan, Y. Zhang, Y. Lu, and Y. Chen, "An efficient chaos-based image compression and encryption scheme using block compressive sensing and elementary cellular automata," *Neural Comput. Appl.*, vol. 32, no. 9, pp. 4961–4988, Nov. 2018.

[21] P. Li and K.-T. Lo, "Joint image encryption and compression schemes based on 16 × 16 DCT," *J. Vis. Commun. Image Represent.*, vol. 58, pp. 12–24, Jan. 2019.

[22] N. Zhou, S. Pan, S. Cheng, and Z. Zhou, "Image compression–encryption scheme based on hyper-chaotic system and 2D compressive sensing," *Opt. Laser Technol.*, vol. 82, pp. 121–133, Aug. 2016.

[23] X. Wu, D. Wang, J. Kurths, and H. Kan, "A novel lossless color image encryption scheme using 2D DWT and 6D hyperchaotic system," *Inf. Sci.*, vols. 349–350, pp. 137–153, Jul. 2016.

[24] X. Chai, X. Zheng, Z. Gan, D. Han, and Y. Chen, "An image encryption algorithm based on chaotic system and compressive sensing," *Signal Process.*, vol. 148, pp. 124–144, Jul. 2018.

[25] L. Gong, C. Deng, S. Pan, and N. Zhou, "Image compression-encryption algorithms by combining hyper-chaotic system with discrete fractional random transform," *Opt. Laser Technol.*, vol. 103, pp. 48–58, Jul. 2018.

[26] H. Huang, X. He, Y. Xiang, W. Wen, and Y. Zhang, "A compression-diffusion-permutation strategy for securing image," *Signal Process.*, vol. 150, pp. 183–190, Sep. 2018.

[27] T. Hu, Y. Liu, L.-H. Gong, S.-F. Guo, and H.-M. Yuan, "Chaotic image cryptosystem using DNA deletion and DNA insertion," *Signal Process.*, vol. 134, pp. 234–243, May 2017.

[28] L. Liu, Q. Zhang, and X. Wei, "A RGB image encryption algorithm based on DNA encoding and chaos map," *Comput. Electr. Eng.*, vol. 38, no. 5, pp. 1240–1248, Sep. 2012.

[29] X.-Y. Wang, Y.-Q. Zhang, and X.-M. Bao, "A novel chaotic image encryption scheme using DNA sequence operations," *Opt. Lasers Eng.*, vol. 73, pp. 53–61, Oct. 2015.

[30] H. Liu, X. Wang, and A. Kadir, "Image encryption using DNA complementary rule and chaotic maps," *Appl. Soft Comput.*, vol. 12, no. 5, pp. 1457–1466, May 2012.

[31] X.-Y. Wang, P. Li, Y.-Q. Zhang, L.-Y. Liu, H. Zhang, and X. Wang, "A novel color image encryption scheme using DNA permutation based on the lorenz system," *Multimedia Tools Appl.*, vol. 77, no. 5, pp. 6243–6265, Mar. 2018.

[32] X. Chai, Z. Gan, K. Yang, Y. Chen, and X. Liu, "An image encryption algorithm based on the memristive hyperchaotic system, cellular automata and DNA sequence operations," *Signal Process., Image Commun.*, vol. 52, pp. 6–19, Mar. 2017.

[33] J. Wu, X. Liao, and B. Yang, "Image encryption using 2D Hénon-sine map and DNA approach," *Signal Process.*, vol. 153, pp. 11–23, Dec. 2018.

[34] J. Chen, Z.-L. Zhu, L.-B. Zhang, Y. Zhang, and B.-Q. Yang, "Exploiting self-adaptive permutation–diffusion and DNA random encoding for secure and efficient image encryption," *Signal Process.*, vol. 142, pp. 340–353, Jan. 2018.

[35] W. Wen, K. Wei, Y. Zhang, Y. Fang, and M. Li, "Colour light field image encryption based on DNA sequences and chaotic systems," *Nonlinear Dyn.*, vol. 99, no. 2, pp. 1587–1600, Jan. 2020.

[36] 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.

[37] Q. Zhang, L. Liu, and X. Wei, "Improved algorithm for image encryption based on DNA encoding and multi-chaotic maps," *AEU Int. J. Electron. Commun.*, vol. 68, no. 3, pp. 186–192, Mar. 2014.

[38] L. Liu, D. Wang, and Y. Lei, "An image encryption scheme based on hyper chaotic system and DNA with fixed secret keys," *IEEE Access*, vol. 8, pp. 46400–46416, 2020.

[39] Y.-Q. Zhang, X.-Y. Wang, J. Liu, and Z.-L. Chi, "An image encryption scheme based on the MLNCML system using DNA sequences," *Opt. Lasers Eng.*, vol. 82, pp. 95–103, Jul. 2016.

[40] X. Wu, H. Kan, and J. Kurths, "A new color image encryption scheme based on DNA sequences and multiple improved 1D chaotic maps," *Appl. Soft Comput.*, vol. 37, pp. 24–39, Dec. 2015.

[41] M. S. Tavazoei, "Fractional order chaotic systems: History, achievements, applications, and future challenges," *Eur. Phys. J. Special Topics*, vol. 229, nos. 6–7, pp. 887–904, Mar. 2020.

[42] F. Yang, J. Mou, K. Sun, Y. Cao, and J. Jin, "Color image compression-encryption algorithm based on fractional-order memristor chaotic circuit," *IEEE Access*, vol. 7, pp. 58751–58763, 2019.

[43] S. He, K. Sun, and H. Wang, "Complexity analysis and DSP implementation of the fractional-order lorenz hyperchaotic system," *Entropy*, vol. 17, no. 12, pp. 8299–8311, Dec. 2015.

[44] Y. Xu, K. Sun, S. He, and L. Zhang, "Dynamics of a fractional-order simplified unified system based on the adomian decomposition method," *Eur. Phys. J. Plus*, vol. 131, no. 6, p. 186, Jun. 2016.

[45] L. Zhang, K. Sun, S. He, H. Wang, and Y. Xu, "Solution and dynamics of a fractional-order 5-D hyperchaotic system with four wings," *Eur. Phys. J. Plus*, vol. 132, no. 1, p. 31, Jan. 2017.

[46] J. Ruan, K. Sun, J. Mou, S. He, and L. Zhang, "Fractional-order simplest memristor-based chaotic circuit with new derivative," *Eur. Phys. J. Plus*, vol. 133, no. 1, p. 3, Jan. 2018.

[47] S. He, K. Sun, and H. Wang, "Solution and dynamics analysis of a fractional-order hyperchaotic system," *Math. Methods Appl. Sci.*, vol. 39, no. 11, pp. 2965–2973, Jul. 2016.

[48] P. Li, J. Xu, J. Mou, and F. Yang, "Fractional-order 4D hyper-chaotic memristive system and application in color image encryption," *EURASIP J. Image Video Process.*, vol. 2019, no. 1, pp. 1–11, Dec. 2019.

[49] Y. Yang, B. Guan, J. Li, D. Li, Y. Zhou, and W. Shi, "Image compression-encryption scheme based on fractional order hyper-chaotic systems combined with 2D compressed sensing and DNA encoding," *Opt. Laser Technol.*, vol. 119, Nov. 2019, Art. no. 105661, doi: 10.1016/j.optlastec.2019.105661.
[50] F. Yang, J. Mou, J. Liu, C. Ma, and H. Yan, "Characteristic analysis of the fractional-order hyperchaotic complex system and its image encryption application," *Signal Process.*, vol. 169, Apr. 2020, Art. no. 107373, doi: 10.1016/j.sigpro.2019.107373.
[51] Y.-Q. Zhang, X.-Y. Wang, L.-Y. Liu, Y. He, and J. Liu, "Spatiotemporal chaos of fractional order logistic equation in nonlinear coupled lattices," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 52, pp. 52–61, Nov. 2017.
[52] R. Montero-Canela, E. Zambrano-Serrano, E. I. Tamariz-Flores, J. M. Muñoz-Pacheco, and R. Torrealba-Meléndez, "Fractional chaos based-cryptosystem for generating encryption keys in ad hoc networks," *Ad Hoc Netw.*, vol. 97, Feb. 2020, Art. no. 102005.
[53] M. A. B. Farah, R. Guesmi, A. Kachouri, and M. Samet, "A novel chaos based optical image encryption using fractional Fourier transform and DNA sequence operation," *Opt. Laser Technol.*, vol. 121, Jan. 2020, Art. no. 105777.
[54] Y. Aydin and F. Ozkaynak, "A provable secure image encryption schema based on fractional order chaotic systems," in *Proc. 23rd Int. Conf. Electron.*, Palanga, Lithuania, Jun. 2019, pp. 1–5.
[55] L. Chen, H. Yin, T. Huang, L. Yuan, S. Zheng, and L. Yin, "Chaos in fractional-order discrete neural networks with application to image encryption," *Neural Netw.*, vol. 125, pp. 174–184, May 2020.
[56] S. M. Ismail, L. A. Said, A. G. Radwan, A. H. Madian, and M. F. Abu-ElYazeed, "A novel image encryption system merging fractional-order edge detection and generalized chaotic maps," *Signal Process.*, vol. 167, Feb. 2020, Art. no. 107280.
[57] F. Ozkaynak, "A novel random number generator based on fractional order chaotic Chua system," *Elektronika ir Elektrotechnika*, vol. 26, no. 1, pp. 52–57, Feb. 2020.
[58] M. Lahdir, H. Hamiche, S. Kassim, M. Tahanout, K. Kemih, and S.-A. Addouche, "A novel robust compression-encryption of images based on SPIHT coding and fractional-order discrete-time chaotic system," *Opt. Laser Technol.*, vol. 109, pp. 534–546, Jan. 2019.
[59] S. S. Jamal, T. Shah, A. H. AlKhaldi, and M. N. Tufail, "Construction of new substitution boxes using linear fractional transformation and enhanced chaos," *Chin. J. Phys.*, vol. 60, pp. 564–572, Aug. 2019.
[60] H. Wang, D. Xiao, X. Chen, and H. Huang, "Cryptanalysis and enhancements of image encryption using combination of the 1D chaotic map," *Signal Process.*, vol. 144, pp. 444–452, Mar. 2018.
[61] Y. Liu, J. Tang, and T. Xie, "Cryptanalyzing a RGB image encryption algorithm based on DNA encoding and chaos map," *Opt. Laser Technol.*, vol. 60, pp. 111–115, Aug. 2014.
[62] J. Chen, L. Chen, and Y. Zhou, "Cryptanalysis of a DNA-based image encryption scheme," *Inf. Sci.*, vol. 520, pp. 130–141, May 2020.
[63] G. Chen and T. Ueta, "Yet another chaotic attractor," *Int. J. Bifurcation Chaos*, vol. 9, no. 7, pp. 1465–1466, Jul. 1999.
[64] X. Ye, J. Mou, Z. Wang, P. Li, and C. Luo, "Dynamic characteristic analysis for complexity of continuous chaotic systems based on the algorithms of SE complexity and C_0 complexity," in *Proc. MLICOM*, Weihai, China, Aug. 2017, pp. 647–657.
[65] D. Cafagna and G. Grassi, "Bifurcation and chaos in the fractional Chua and Chen systems with very low order," in *Proc. IEEE Int. Symp. Circuits Syst.*, Taipei, Taiwan, May 2009, pp. 2846–2849.
[66] X. Wu, Y. Li, and J. Kurths, "A new color image encryption scheme using CML and a fractional-order chaotic system," *PLoS ONE*, vol. 10, no. 3, Mar. 2015, Art. no. 0119660.
[67] N. T. Shawagfeh, "Analytical approximate solutions for nonlinear fractional differential equations," *Appl. Math. Comput.*, vol. 131, nos. 2–3, pp. 517–529, Sep. 2002.
[68] G. Adomian, "A new approach to nonlinear partial differential equations," *J. Math. Anal. Appl.*, vol. 102, pp. 420–434, Sep. 1984.
[69] R. M. May, "Simple mathematical models with very complicated dynamics," *Nature*, vol. 261, no. 5560, pp. 459–467, Jun. 1976.
[70] X.-J. Tong, M. Zhang, Z. Wang, and J. Ma, "A joint color image encryption and compression scheme based on hyper-chaotic system," *Nonlinear Dyn.*, vol. 84, no. 4, pp. 2333–2356, Feb. 2016.
[71] Y.-Q. Zhang and X.-Y. Wang, "A symmetric image encryption algorithm based on mixed linear–nonlinear coupled map lattice," *Inf. Sci.*, vol. 273, pp. 329–351, Jul. 2014.
[72] F. Özkaynak, "Brief review on application of nonlinear dynamics in image encryption," *Nonlinear Dyn.*, vol. 92, no. 2, pp. 305–313, Jan. 2018.
[73] J. Chen, Y. Zhang, L. Qi, C. Fu, and L. Xu, "Exploiting chaos-based compressed sensing and cryptographic algorithm for image encryption and compression," *Opt. Laser Technol.*, vol. 99, pp. 238–248, Feb. 2018.

HAO DONG received the B.E. degree in engineering from Jilin Beihua University, in 2019. He is currently pursuing the M.E. degree in information and communication engineering with Donghua University, Shanghai, China. His current research interests are image encryption and chaos theory.

ENJIAN BAI received the B.S. degree in mathematics from Qufu Normal University, and the M.S. and Ph.D. degrees in cryptography from Xidian University. He is currently an Associate Professor with the College of Information Science and Technology, Donghua University, Shanghai, China. His mainly research interests are in applied mathematics, cryptography, and fuzzy systems.

XUE-QIN JIANG received the B.S. degree in computer science from the Nanjing Institute of Technology, Nanjing, China, and the M.S. and Ph.D. degrees in electronics engineering from Chonbuk National University, Jeonju, South Korea. He is currently a Full Professor with the School of Information Science and Technology, Donghua University, Shanghai, China. He has authored or coauthored more than 50 SCI articles. His main research interests include wireless communications, physical-layer security, and channel coding. He is currently serving as an Editor of the IEEE COMMUNICATIONS LETTERS and IEEE ACCESS.

YUN WU received the B.S. and M.S. degrees in electrical engineering from the Harbin Institute of Technology, Harbin, China, in 1999 and 2001, respectively, and the Ph.D. degree from Shanghai Jiaotong University, Shanghai, China, in 2006. She is currently an Associate Professor with the School of information Science and Technology, Donghua University. Her research interests include channel estimation and synchronization, cognitive radio technology, MIMO, and compressive sensing.