IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 9, 2021, accepted January 21, 2021, date of publication January 27, 2021, date of current version February 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3054842

An Image Compression and Encryption Algorithm Based on the Fractional-Order Simplest Chaotic Circuit

HAIYING HU, YINGHONG CAO, JI XU, CHENGUANG MA, AND HUIZHEN YAN
School of Information Science and Engineering, Dalian Polytechnic University, Dalian 116034, China

Corresponding author: Yinghong Cao (caoyinghong@dlpu.edu.cn)

This work was supported by the Basic Scientific Research Projects of Colleges and Universities of Liaoning Province under Grant 2017J045.

ABSTRACT Based on compressive sensing and fractional-order simplest memristive chaotic system, this paper proposes an image compression and encryption scheme. First, a fractional-order simplest memristive chaotic circuit system is designed. The dynamic characteristics of the chaotic system are analyzed by the phase diagram, the Lyapunov exponent's spectrum, and the bifurcation diagram to determine the parameters and pseudo-random sequences used in the encryption scheme. Secondly, an encryption scheme based on compressive sensing is designed. This scheme compresses the image twice to fully reduce the storage cost, and scrambles the pixel matrix twice through block scrambling and zigzag transformation, and then uses chaotic pseudo-random sequence and GF (17) domain diffusion image matrix to obtain the final cipher image. Finally, simulation results and performances analysis indicate that the scheme still has good reconstruction performance, even when the compression ratio is 0.25, and the security analysis shows that it can resist various attacks and has high security.

INDEX TERMS Image encryption, compressive sensing (CS), fractional-order simplest memristive chaotic system.

I. INTRODUCTION

The rapid development of communication and Internet technology has made digital images widely used in various fields. Because image information contains privacy and confidentiality, secure real-time transmission of digital images is incredibly important. Therefore, digital images need to be encrypted before transmission to ensure information security. Recently, many algorithms for digital image encryption are presented [1]–[14]. For example, Cao, *et al.* [1] used a 2D-LCMIC hyperchaotic map to design an image encryption scheme that uses both bit-level displacement and diffusion. A novel image cryptosystem based on the tent chaotic map was presented by Li, *et al.* [2]. Zhang, *et al.* [5] introduced an image encryption algorithm by DNA sequence operations and chaos in 2010. Liu, *et al.* [6] proposed a fast image encryption algorithm based on 2D Sine ICMIC modulation map for scrambling and diffusion simultaneously. Wang *et al.* introduced a novel image encryption scheme through DNA sequence operations and 2D logistic chaotic system [7]. Zhang, *et al.* [8] proposed a color image encryption algorithm that uses a hybrid model of bidirectional cyclic substitution and DNA sequence manipulation. Liu, *et al.* [15] designed a stream-cipher algorithm based on one-time key using piecewise linear chaotic map. Wang, *et al.* [16] proposed a chaotic image encryption scheme based on a simple perceptron. Liu, *et al.* [17] introduced a novel image encryption algorithm that uses DNA coding for pixel diffusion. The above methods achieve digital image encryption, whereas these algorithms do not compress the image, which are not suitable for digital image transmission under the circumstances of limited storage resources and transmission bandwidth.

To overcome these weaknesses, Candes and Donoho proposed CS theory. If the signal is sparse, random sampling can be used to obtain discrete samples of the signal, and the signal is reconstructed by a non-linear reconstruction algorithm at a condition much lower than the Nyquist sampling

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

rate [18], [19]. With the development of CS, it has been shown to be able to effectively reduce storage and transmission costs [20], [21]. Therefore, the application of compressive sensing to image encryption algorithms has been widely studied. [22]–[34]. Wang, *et al.* [22] designed an image password system using embedding technology and parallel compressive sensing counter mode, but the encrypted images have high correlation, so the security of the algorithm is lower. Chai, *et al.* [23] proposed an image encryption scheme based on the memristive chaotic system, basic cellular automata and compressive sensing. In this algorithm, the key is relevant to the plaintext, which can effectively resist the plaintext attacks. However, the pixel distribution of the encrypted image is uneven, so the image demands to be re-encrypted after compressive sensing. Mou, *et al.* [24] introduced an image encryption scheme, which combined the 3D-SIMM chaotic system and compressive sensing. Xu, *et al.* [25] studied an image cryptosystem, which is based on compressive sensing and 2D-SLIM hyper-chaotic map. Zhou, *et al.* [26] described a 2D compressive sensing image encryption scheme based on the Chen hyperchaotic system. In the above method, hyperchaotic system and compressive sensing are used in image compression and encryption algorithm, which can effectively compress the images and reduce the transmission cost, but those methods are not safe enough and the accuracy of image reconstruction is low when the image compression rate is small.

Fractional calculus has been proposed for more than 300 years. Due to the lack of effective calculation tools, it has not attracted widespread attention. In recent years, with the development of computer science, fractional calculus has become a research hotspot. Research shows that fractional calculus provides new mathematical tools and theories for studying some complex phenomena and systems. When building a chaotic system with strong chaotic characteristics, scholars found that after introducing fractional calculus, not only the degree of freedom of the system increases, but the chaotic characteristics of the system also become more complicated. It is found that compared with other chaotic systems, fractional-order systems have richer dynamic characteristics. Because fractional derivatives are non-local and highly non-linear, their geometric interpretation is very complex [35]–[40]. In addition, applying fractional order to chaotic systems can increase the key space of the system to improve the reliability of the cryptosystem. Therefore, some encryption schemes combining fractional-order chaotic systems are proposed [41]–[50]. Yang, *et al.* [41] studied a color image compression-encryption scheme based on fractional-order memristive chaotic map, but the memristive chaotic circuit is complicated. Hence, it is difficult to achieve in practical applications. A symmetric digital image encryption scheme based on an improper fractional-order chaotic system was proposed by Zhao, *et al.* [42]. However, the information entropy is not close to the ideal value of 8, so the security performance is not high. Huang, *et al.* [43] studied an image password system based on fractional-order hyperchaotic map. However, a small key space is not effective against brute-force attacks.

To further improve the security and the image reconstruction accuracy, an image encryption algorithm based on compressive sensing and fractional-order simplest memristive chaotic system is proposed. First, the Adomian decomposition method solves the fractional-order simplest memristive chaotic system to determine the parameters, and applies the chaotic sequence generated by the system to the entire encryption algorithm. Secondly, the discrete cosine transform is performed on the plain image to obtain the sparse coefficient matrix. Then the sparse coefficient matrix is compressed and sampled by the measurement matrix generated by the Hadamard matrix and the chaotic pseudo-random sequence. This compression is performed twice to reduce the transmission cost sufficiently. Finally, the compressed image is encrypted by block scrambling, zigzag transform and GF (17) diffusion to ensure the algorithm has good security performance.

The rest parts of the paper are organized as follows. In Section 2, the fractional-order simplest memristive chaotic system is designed and compressive sensing theory is presented. In Section 3, an image compression and encryption process based on CS is described. The proposed algorithm is simulated and its performance is analyzed in Section 4. Finally came to a conclusion.

II. COMPRESSIVE SENSING AND FRACTIONAL-ORDER SIMPLEST MEMRISTIVE CHAOTIC SYSTEM

This section introduces the principle of compressive sensing, and then uses the ADM algorithm to solve the fractional-order simplest memristive chaotic system, and analyzes the dynamic characteristics of the chaotic system through the phase diagram, Lyapunov exponent's spectrum, and bifurcation. From this, determines the parameter values and chaotic pseudo-random sequences used in the encryption scheme.

A. COMPRESSIVE SENSING

Compressive sensing exploits signal sparsity. If the signal is sparse, random sampling can be used to obtain discrete samples of the signal, and the signal is reconstructed by a nonlinear reconstruction algorithm at a condition much lower than the Nyquist sampling rate. Its processing flow is shown in Fig. 1.

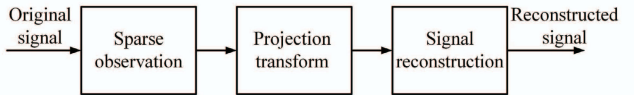


FIGURE 1. Compressive sensing processing flow.

Assuming that a one-dimensional signal $X \in R^M$ of length M, Which can adopt an $M \times M$ dimensional orthogonal basis matrix, and a linear combination of $\psi = [\psi_1, \psi_2, \psi_3, \cdots, \psi_M]$ is expressed as follows:

$$x \in \sum_{j=1}^{n} \alpha_j \psi_j = \psi\alpha \tag{1}$$

where ψ_j is the column vector, α_j is the weighting coefficient, and the prerequisite for compressive sensing of signal x is that the signal is sparse. If there are L ($L \ll M$) non-zero values in α, then x is an L sparse signal. Projecting the signal x with the measurement matrix Φ as follows:

$$y = \Phi x = \Phi\psi\alpha \tag{2}$$

This is the dimension reduction process. By solving the l_0-norm minimization problem, the signal x is reconstructed from

$$\widehat{\alpha} = \text{argmin}||\alpha||_0, s.t. y = \Phi\psi\alpha \tag{3}$$

where $||\cdot||_0$ represents the l_0-norm of a vector.

In this encryption scheme, the plaintext image pixels are sparse by DCT transformation, the measurement matrix is generated by the chaotic pseudo-random sequences and the Hadamard matrix. Meanwhile, the image is reconstructed by the orthogonal matching pursuit (OMP) algorithm.

B. FRACTIONAL-ORDER SIMPLEST MEMRISTIVE CHAOTIC SYSTEM

1) ADOMIAN DECOMPOSITION METHOD

For a given fractional differential equation ${}^*D^q_{t_0}(t) = f(x(t))$, its function variable $x(t) = [x_1(t), x_2(t), x_3(t), \cdots, x_n(t)]^T$ is the state variable, and ${}^*D^q_{t_0}$ represents the Caputo differential operator of order q, where $m-1 < q \leq m$, $m \in N$. The function $f(x(t))$ is divided into linear, nonlinear and constant terms.

$$\begin{cases} {}^*D^q_{t_0}x(t) = Lx(t) + Nx(t) + g(t) \\ x^{(k)}(t_0^+) = b_k, k = 0, 1, \ldots, m-1 \\ g(t) = [g_1(t), g_2(t), \ldots, g_n(t)]^T \end{cases} \tag{4}$$

where L is the linear term, N is the nonlinear term, $g(t)$ is the system constant and b_k is the initial value. After applying operator $J^q_{t_0}$ to both sides of equation (4), it can be obtained from

$$x = J^q_{t_0}Lx + J^q_{t_0}Nx + J^q_{t_0}g + \sum_{k=0}^{m-1} b_k \frac{(t-t_0)^k}{k!} \tag{5}$$

here, $J^q_{t_0}$ is the R-L fractional integral operator of order q, the fractional integral operator has the following basic properties:

$$J^q_{t_0}(t-t_0)^\gamma = \frac{\Gamma(\gamma+1)}{\Gamma(\gamma+1+q)}(t-t_0)^{\gamma+q} \tag{6}$$

$$J^q_{t_0}C = \frac{C}{\Gamma(q+1)}(t-t_0)^q \tag{7}$$

$$J^q_{t_0}J^r_{t_0}x(t) = J^{q+r}_{t_0}x(t) \tag{8}$$

where $t \in [t_0, t_1]$, $\gamma \geq -1$, $q \geq 0$, $r \geq 0$, C is a constant term. According to the ADM decomposition algorithm, the nonlinear terms in Eq. (5) can be decomposed according to the Eq. (9)

$$\begin{cases} A^i_j = \frac{1}{i!}[\frac{d^i}{d\lambda^i}N(\nu^i_j(\lambda))]_{\lambda=0} \\ \nu^i_j(\lambda) = \sum_{k=0}^{i}(\lambda)^k x^k_j \end{cases} \tag{9}$$

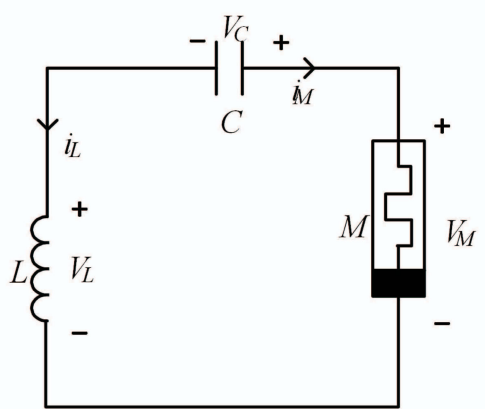


FIGURE 2. Simplest memristive chaotic circuit.

here $i = 0, 1, 2, 3, \cdots, \infty$, $j = 1, 2, 3, \cdots, n$. Then the nonlinear part is

$$Nx = \sum_{i=0}^{\infty} A^i(x^0, x^1, \ldots, x^i) \tag{10}$$

Therefore, the numerical solution of Eq. (4) is

$$\begin{cases} x^0 = J^q_{t_0}g + \sum_{k=0}^{m-1} b_k \frac{(t-t_0)^k}{k!} \\ x^1 = J^q_{t_0}Lx^0 + J^q_{t_0}A^0(x^0) \\ x^2 = J^q_{t_0}Lx^1 + J^q_{t_0}A^1(x^0, x^1) \\ \ldots \\ x^i = J^q_{t_0}Lx^{i-1} + J^q_{t_0}A^{i-1}(x^0, x^1, \ldots, x^{i-1}) \\ \ldots \end{cases} \tag{11}$$

2) FRACTIONAL-ORDER SIMPLEST MEMRISTIVE CHAOTIC CIRCUIT

The fractional-order simplest memristive chaotic circuit model consisting of a nonlinear active charge-controlled memristor, a linear passive capacitor and a linear passive inductor is shown in Fig. 1. M, V_M, V_L, V_C, i_M and i_L represent the state variables of the circuit.

According to the volt-ampere characteristics of each element and Kirchhoff's law, the simplest memristive chaotic circuit can be determined by the first-order differential Eq. (12).

$$\begin{cases} C\frac{dV_C(t)}{dt} = i_L(t) \\ L\frac{di_L(t)}{dt} = -(V_C(t) + \beta(z^2(t)-1)i_L(t)) \\ \frac{dz(t)}{dt} = i_L(t) - \alpha z(t) + i_L(t)z(t) \end{cases} \tag{12}$$

Let $V_C(t) = x$, $i_L(t) = y$, $z(t) = z$, $1/C = a$, $1/L = b$, the normalization operation of equation (12) are:

$$\begin{cases} \dot{x} = ay \\ \dot{y} = -b(x + \beta(z^2-1)y) \\ \dot{z} = -y - \alpha z + yz \end{cases} \tag{13}$$

here a, b, α, β are the system parameters, fixed parameters $a = 1$, $b = 1/3$, $\alpha = 0.6$, $\beta = 1.5$, and the initial values of Eq. (13) are [0.1, 0, 0]. By calculating the Lyapunov exponents of the system are $L_1 = 0.0453$, $L_2 = 0$, $L_3 = -0.6025$. Since the system has a positive Lyapunov exponent and the Lyapunov dimension $D_L = 2.075$ is the score. It indicates that the system is in chaotic state.

From the definition of Caputo fractional calculus and the simplest memristive chaotic circuit system equation (13), the mathematical expression of the fractional-order simplest memristive chaotic system is

$$\begin{cases} {}_{*}D^{q}_{t_0}x = ay \\ {}_{*}D^{q}_{t_0}y = -b(x + \beta(z^2 - 1)y) \\ {}_{*}D^{q}_{t_0}z = -y - \alpha z + yz \end{cases} \tag{14}$$

where x, y, z are the state variables, ${}_{*}D^{q}_{t_0}$ is the Caputo operator, t_0 is the integral initial value, $0 < q \leq 1$ is the order of fractional-order system equation, α and β are internal parameters of the memristor, a and b are the system parameters.

3) NUMERICAL SOLUTION OF FRACTION-ORDERS SIMPLEST MEMRISTIVE CHAOTIC CIRCUIT

Using the ADM decomposition method, the linear and nonlinear terms of Eq. (14) are obtained from

$$\begin{bmatrix} Lx_1 \\ Lx_2 \\ Lx_3 \end{bmatrix} = \begin{bmatrix} ax_2 \\ -bx_1 - b\beta x_2 \\ -x_2 + \alpha x_3 \end{bmatrix}$$
$$\begin{bmatrix} Nx_1 \\ Nx_2 \\ Nx_3 \end{bmatrix} = \begin{bmatrix} 0 \\ -b\beta x_3^2 x_2 \\ x_2 x_3 \end{bmatrix} \tag{15}$$

According to Eq. (9), the first six ADM polynomials for nonlinear terms $x_2x_3x_3$ and x_2x_3 are respectively decomposed into:

$$\begin{cases} A_2^0 = x_2^0(x_3^0)^2 \\ A_2^1 = x_2^1(x_3^0)^2 + 2x_2^0x_3^1x_3^0 \\ A_2^2 = x_2^2(x_3^0)^2 + 2x_2^1x_3^1x_3^0 + x_2^0(x_3^1)^2 + 2x_2^0x_3^2x_3^0 \\ A_2^3 = x_2^3(x_3^0)^2 + 2x_2^2x_3^1x_3^0 + x_2^1(x_3^1)^2 + 2x_2^1x_3^2x_3^0 \\ \quad +2x_2^0x_3^2x_3^1 + 2x_2^0x_3^3x_3^0 \\ A_2^4 = x_2^4(x_3^0)^2 + 2x_2^3x_3^1x_3^0 + x_2^2(x_3^1)^2 + 2x_2^2x_3^2x_3^0 \\ \quad +2x_2^1x_3^2x_3^1 + 2x_2^1x_3^3x_3^0 + 2x_2^0x_3^4x_3^0 + 2x_2^0x_3^1x_3^3 \\ \quad +x_2^0(x_3^2)^2 \\ A_2^5 = x_2^5(x_3^0)^2 + 2x_2^4x_3^1x_3^0 + x_2^3(x_3^1)^2 + 2x_2^3x_3^2x_3^0 \\ \quad +2x_2^2x_3^2x_3^1 + 2x_2^2x_3^3x_3^0 + 2x_2^1x_3^4x_3^0 + 2x_2^1x_3^1x_3^3 \\ \quad +x_2^1(x_3^2)^2 + 2x_2^0x_3^4x_3^1 + 2x_2^0x_3^2x_3^3 + 2x_2^0x_3^5x_3^0 \end{cases} \tag{16}$$

$$\begin{cases} A_3^0 = x_2^0x_3^0 \\ A_3^1 = x_2^1x_3^0 + x_2^0x_3^1 \\ A_3^2 = x_2^2x_3^0 + x_2^1x_3^1 + x_2^0x_3^2 \\ A_3^3 = x_2^3x_3^0 + x_2^2x_3^1 + x_2^1x_3^2 + x_2^0x_3^3 \\ A_3^4 = x_2^4x_3^0 + x_2^3x_3^1 + x_2^2x_3^2 + x_2^1x_3^3x + x_2^0x_3^4 \\ A_3^5 = x_2^5x_3^0 + x_2^4x_3^1 + x_2^3x_3^2 + x_2^2x_3^3 + x_2^1x_3^4 + x_2^0x_3^5 \end{cases} \tag{17}$$

According to the equations (6)-(8) and (11), the discrete iterative calculation formula of the system (14) can be obtained as:

$$\begin{cases} x_{m+1} = x_m + ay_m\dfrac{h^q}{\Gamma(q+1)} + ab(x_m - \beta y_m + \beta y_m z_m^2) \\ \qquad \dfrac{h^{2q}}{\Gamma(2q+1)} + \ldots \\ y_{m+1} = y_m + b(x_m - \beta y_m + \beta y_m z_m^2)\dfrac{h^q}{\Gamma(q+1)} \\ \qquad +(aby_m - b^2\beta x_m + \ldots)\dfrac{h^{2q}}{\Gamma(2q+1)} + \ldots \\ z_{m+1} = z_m + (y_m z_m - y_m - \alpha z_m)\dfrac{h^q}{\Gamma(q+1)} + (z_m b x_m \\ \qquad +\ldots)\quad \dfrac{h^{2q}}{\Gamma(2q+1)} + \ldots \end{cases} \tag{18}$$

where h is the iteration step size, $\Gamma(\cdot)$ is the gamma function. The iterative algorithm in the simulation process is (19)-(25).

$$\begin{cases} C_{10} = x_m \\ C_{20} = y_m \\ C_{30} = z_m \end{cases} \tag{19}$$

$$\begin{cases} C_{11} = aC_{20} \\ C_{21} = bC_{10} - b\beta C_{20} + b\beta C_{20}C_{30}^2 \\ C_{30} = -C_{20} - \alpha C_{30} + C_{20}C_{30} \end{cases} \tag{20}$$

$$\begin{cases} C_{12} = aC_{21} \\ C_{22} = bC_{11} - b\beta C_{21} + b\beta(C_{21}C_{30}^2 \\ \qquad +2C_{20}C_{30}C_{31}) \\ C_{32} = -C_{21} - \alpha C_{31} + C_{21}C_{30} + C_{20}C_{31} \end{cases} \tag{21}$$

$$\begin{cases} C_{13} = aC_{22} \\ C_{23} = bC_{12} - b\beta C_{22} + b\beta(C_{22}C_{30}^2 + 2C_{20}C_{30}C_{32}) \\ \qquad +b\beta(2C_{21}C_{30}C_{21} + C_{20}C_{31}^2)\dfrac{\Gamma(2q+1)}{\Gamma^2(q+1)} \\ C_{33} = -C_{22} - \alpha C_{32} + C_{22}C_{30} + C_{20}C_{32} + C_{21}C_{31} \\ \qquad \dfrac{\Gamma(2q+1)}{\Gamma^2(q+1)} \end{cases} \tag{22}$$

$$\begin{cases} C_{14} = aC_{23} \\ C_{24} = bC_{13} - b\beta C_{23} + b\beta(C_{23}C_{30}^2 + 2C_{20}C_{30}C_{33} \\ \qquad +(2C_{22}C_{30}C_{31} + 2C_{21}C_{30}C_{32} + 2C_{20}C_{31}C_{32}) \\ \qquad \dfrac{\Gamma(3q+1)}{\Gamma(q+1)\Gamma(2q+1)} + C_{21}C_{31}^2\dfrac{\Gamma(3q+1)}{\Gamma^3(q+1)}) \\ C_{34} = -C_{23} - \alpha C_{33} + C_{23}C_{30} + C_{20}C_{33} + (C_{21}C_{32} \\ \qquad +C_{22}C_{31})\dfrac{\Gamma(3q+1)}{\Gamma(q+1)\Gamma(2q+1)} \end{cases} \tag{23}$$

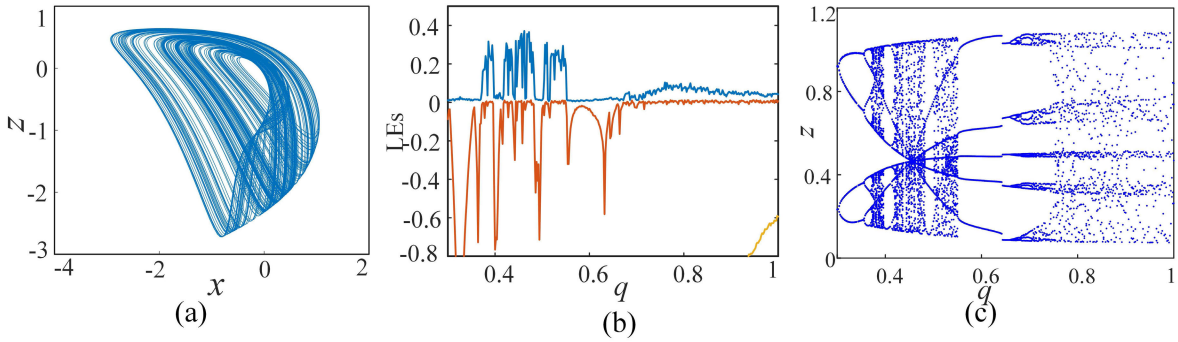


FIGURE 3. Dynamics of the chaotic map: (a) $x-z$ plane phase diagram (b) Lyapunov exponents spectrum of q (c) Bifurcation diagram of q.

$$\begin{cases} C_{15} = aC_{24} \\ C_{25} = bC_{14} - b\beta C_{24} + b\beta(C_{24}C_{30}^2 + 2C_{20}C_{30}C_{34} \\ \quad +(2C_{23}C_{30}C_{31} + 2C_{21}C_{30}C_{33} + 2C_{20}C_{31}C_{33}) \\ \quad \frac{\Gamma(4q+1)}{\Gamma(q+1)\Gamma(3q+1)} + (2C_{22}C_{30} \\ \quad +C_{20}C_{32}^2)\frac{\Gamma(4q+1)}{\Gamma^2(2q+1)} \\ \quad +(C_{22}C_{31}^2 + 2C_{21}C_{31}C_{32})\frac{\Gamma(4q+1)}{\Gamma(2q+1)\Gamma^2(q+1)}) \\ C_{35} = -C_{24} - \alpha C_{34} + C_{24}C_{30} + C_{20}C_{34} + (C_{23}C_{31} \\ \quad +C_{21}C_{33})\frac{\Gamma(4q+1)}{\Gamma(q+1)\Gamma(3q+1)} \\ \quad +C_{22}C_{32}\frac{\Gamma(4q+1)}{\Gamma^2(2q+1)} \end{cases} \tag{24}$$

Therefore the solution of system (14) is defined as

$$x_j(t) = \sum_{i=0}^{5} c_j^i \frac{(t-t_0)^{iq}}{i!q^i} \tag{25}$$

where $h = t - t_0$ is time step, $j = 1, 2, 3$.

4) SYSTEM DYNAMICS ANALYSIS

Setting the parameters $\alpha = 0.6$, $\beta = 1.5$, $a = 1$, $b = 1/3$, $q = 0.545$, $h = 0.01$, and the initial values $[x_0, y_0, z_0] = [0.1, 0, 0]$. Then the chaotic attractor phase diagram can be obtained as Fig. 3(a). The Lyapunov exponents are $L_1 = 0.3459$, $L_2 = 0$, $L_3 = -8.1652$, and Lyapunov dimension $D_L = 2.042$. There is a Lyapunov exponent greater than zero in these Lyapunov exponent values, so the system is in chaotic state. The Lyapunov exponent's spectrum and the bifurcation diagram are important indicators for assessing the dynamical behaviors of chaotic systems. Fig. 3(b) and (c) are the Lyapunov exponents spectrum and bifurcation diagram of $q \in (0.3, 1)$ respectively, the minimum order for generating chaos was observed as $q = 0.38 \times 3.1.14$. It is obviously that the fractional-order simplest chaotic system is highly random, highly sensitive of initial values and parameters, and the system can generate more random chaotic sequences, which can effectively improve the security of encrypted images.

TABLE 1. Test results of NIST test.

Test Name	Our chaotic system		Ref. [41]	
	P-value	Pass rate	P-value	Pass rate
Frequency	0.275709	0.98	0.867692	0.99
Block Frequency	0.224821	0.98	0.77918	1
Cumulative Sums	0.275709	0.98	0.739918	0.99
Runs	0.249284	0.99	0.779188	0.98
Longest Run	0.350485	0.99	0.055361	1
Rank	0.055361	0.98	0.474986	0.99
FFT	0.851383	1	0.062821	1
Non Overlapping Template	0.508088	0.99	0.071177	0.99
Overlapping Template	0.637119	0.98	0.013569	0.99
Universal	0.262249	1	0.108791	0.99
Approximate Entropy	0.304126	0.99	0.759756	1
Random Excursions	0.368150	0.98	0.249284	0.98
Random Excursions Variant	0.348970	0.98	0.025193	0.98
Serial	0.397096	0.99	0.137282	0.98
Linear Complexity	0.071177	0.99	0.227821	0.97

5) THE RANDOMNESS OF FRACTIONAL-ORDER SIMPLEST MEMRISTIVE CHAOTIC SYSTEM

In order to quantitatively analyze the pseudo-randomness of the chaotic sequence, the NIST SP 800-22 test package is used to test the randomness of the sequences. The software package uses 15 performance indicators and 2 judgment criteria (P-value, pass rate) to evaluate the randomness of chaotic sequences.

P-value reflects the uniform distribution of the chaotic sequence, which is calculated as

$$\chi^2 = \sum_{i=1}^{10} \frac{(F_i - 0.1m)^2}{0.1m} P - value = \text{igmac}(4.5, \frac{\chi^2}{2}) \tag{26}$$

where F_i represents the number of P-value between $(0.1(i\text{-}1), 0.1i)$, m represents the number of groups, and igamc is a high-priced incomplete gamma function. If the P-value is greater than 0.0001, it means the sequence is random.

The pass rate is mainly the percentage of passing the test sequence, and the confidence interval for passing the test is

$$1 - \alpha \pm \sqrt{\frac{\alpha(1-\alpha)}{m}} \tag{27}$$

where the significance level α is 0.01, $m \geq 1000$. Table 1 lists the NIST test results of the fractional-order simplest memristive chaotic system and the test results of Ref [41].

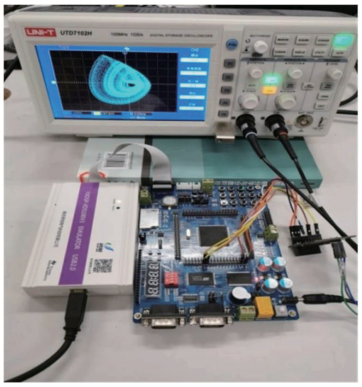


FIGURE 4. DSP implementation platform.

6) DSP REALIZATION OF FRACTIONAL-ORDER SIMPLEST MEMRISTIVE CHAOTIC SYSTEM

In this part, the fractional-order simplest memristive chaotic system is realized by the DSP platform. It's hardware implementation platform is shown in Figure 4. The DSP chip used here is TMS320F28335, which features fast processing speed, high accuracy and high reliability. A 16-bit dual-channel D/A converter (DAC8552) controlled by SPI converts digital signals into analog signals. The output signal is displayed on the oscilloscope (UTD7102H). First, preprocess the data, and then write the formula (20-24) on the DSP board through C language programming. After DA conversion and output result, it is input to the oscilloscope.

Setting $a = 1$, $b = 1/3$, $\alpha = 0.6$, $\beta = 1.5$, $q = 0.545$, $h = 0.01$, and initial conditions $[x_0, y_0, z_0] = [0.1, 0, 0]$, Fig. 5(a)-(c) show the phase diagram of the fractional-order simplest memristive chaotic map captured by the oscilloscope, which are the same as the computer simulation results.

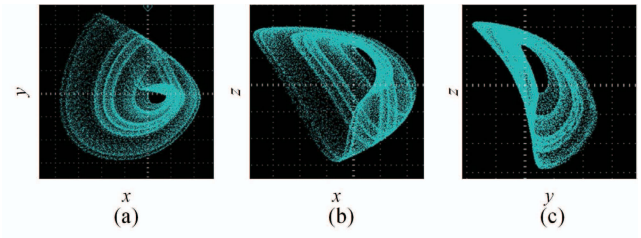


FIGURE 5. Phase diagram realized by DSP platform (a) $x-y$ plane (b) $x-z$ plane (c) $y-z$ plane.

III. IMAGE ENCRYPTION AND DECRYPTION ALGORITHM

A. ZiGZAG TRANSFORM METHOD

The elements of a matrix are scanned in the zigzag order from the upper left corner, as shown in Fig. 6(a). First, the scanned elements are sequentially stored in a one-dimensional array, and then a two-dimensional matrix is generated from the one dimensional array. The process can be recognized from the Fig. 6(b) (the number in the matrix indicates the index of the element value at that position). Therefore, the above transformation process can be regarded as scrambling of elements in the matrix, and this transformation is called zigzag transformation. Since digital images can be represented by a matrix, the above ideas can be used for scrambling digital images.

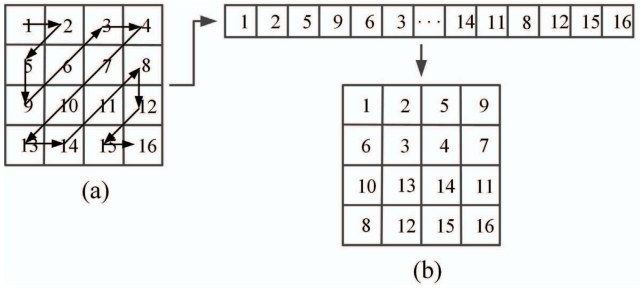


FIGURE 6. Zigzag transformation process (a) Original matrix (b) The matrix after zigzag transformation.

B. IMAGE ENCRYPTION ALGORITHM

The image encryption algorithm flow chart is shown in figure 7. It consists three parts. (1) Sparse and CS calculation processing on the grayscale plane image. (2) To obtain better security, the block scrambling and zigzag transform are used to scramble the compressed image. (3) Using the GF (17) algorithm to diffuse the scrambled image to obtain the final encrypted image.

Step 1: The discrete cosine transform (DCT) is used to transform a digital image P with size of $H \times H$ into a sparse coefficient matrix P_1 of the same size as P.

Step 2: Setting the initial conditions and parameters of the chaotic system, and the system (14) is iterated $m + M$ times to obtain three chaotic sequences, and then the x sequence is combined with the Hadamard matrix to generate the $M \times H$ ($M = H \times CR$) measurement matrix Φ, where CR is the compression ratio and it is defined as equation (28), here, P is original image, C is encrypted image.

$$CR = \frac{C_{\text{heigt}} \times C_{\text{width}}}{P_{\text{heigt}} \times P_{\text{width}}} \tag{28}$$

Step 3: According to the Eq. (29), the image is linearly projected twice to obtain a compressed image P_2 of a size of $M \times M$.

$$P_2 = \Phi(\Phi\psi P)' = \Phi(\Phi P_1)' \tag{29}$$

where ψ is the DCT matrix, P is the original image.

Step 4: The matrix P_2 is quantized to obtain P_3, and the value of P_3 is limited to an integer between 0 and 255.

Step 5: The matrix P_3 is divided into blocks, and since the compression ratio is different, the matrix P_3 is different in size, so P_3 is divided into blocks of different sizes for different compression ratios. The block matrix is scrambled according to Fig. 8.

Step 6: Firstly, the matrices after the block scrambled are combined into a large image matrix, and then the image pixel positions are scrambled by using the zigzag transform depicted in Section 3.1.

Step 7: To obtain the chaotic pseudo-random sequences in the diffusion process, the initial values and parameters of

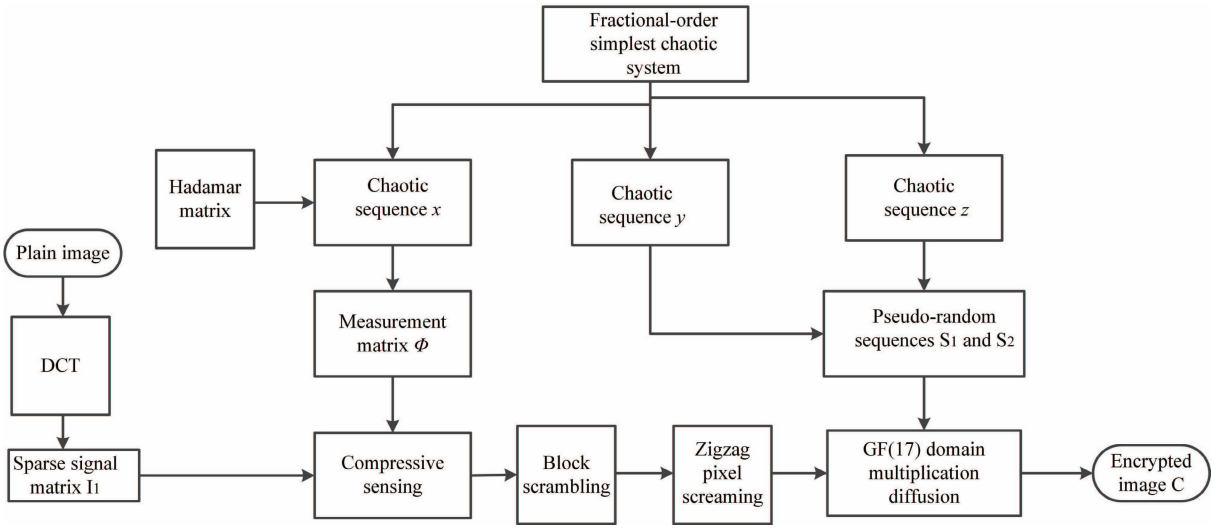


FIGURE 7. Encryption process flow chart.

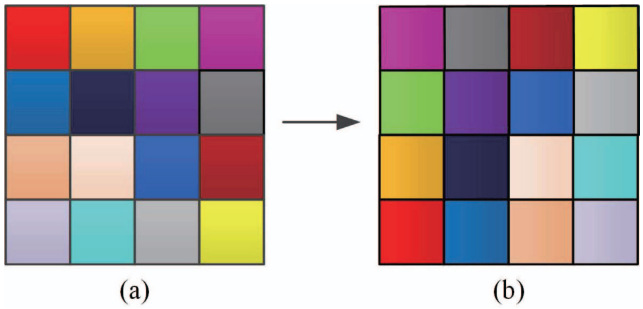


FIGURE 8. Image block scrambling (a) Original matrix block (b) The matrix block after scrambling.

the chaotic system are set, and then the system (14) iterates $N + M \times M$ times, taking the previous $M \times M$ terms to get the sequence y with length $M \times M$ and taking the last $M \times M$ terms to get the sequence z with length $M \times M$, Y and Z are generated from

$$Y = \mathrm{mod}(floor(y * pow2(16)), 256) \tag{30}$$

$$Z = \mathrm{mod}(floor(z * pow2(16)), 256) \tag{31}$$

Step 8: The pseudo-random sequences S_1 and S_2 of forward diffusion and reverse diffusion are obtained by Y and Z, respectively.

Step 9: After the scrambled algorithm, the GF (17) multiplication diffusion algorithm is applied to the gray value of the pixel. This paper uses a combination of forward diffusion and reverse diffusion. Forward and reverse diffusion processes are

$$\begin{cases} C_{i,H} = C_{i-1,H} \times S_{i,H} \times P_{i,H} \\ C_{i,L} = C_{i-1,L} \times S_{i,L} \times P_{i,L} \\ C = (C_{i,H} \times 16 + C_{i,L}) \end{cases} \tag{32}$$

$$\begin{cases} C_{i,H} = C_{i+1,H} \times S_{i,H} \times P_{i,H} \\ C_{i,L} = C_{i+1,L} \times S_{i,L} \times P_{i,L} \\ C = (C_{i,H} \times 16 + C_{i,L}) \end{cases} \tag{33}$$

in which, Eq. (32) and Eq. (33) are the forward diffusion process and the reverse diffusion process, respectively. P represents one-dimensional vector of the pixel matrix. C and S are cryptographic vectors, initial values C_0 comes from the secret key ($i = 1, 2, 3, \cdots, M \times M$), H is the upper 4 bits of the data, and L represents the lower 4 bits of the data.

Step 10: The encrypted image C of size $M \times M$ is obtained by converting the diffused vector into a matrix.

C. IMAGE DECRYPTION ALGORITHM

The image decryption process is illustrated in Fig. 9. This is the reverse process of the encryption algorithm, and image reconstruction uses the OMP algorithm, and the detailed decryption steps are as follows.

Step 1: The encrypted image C is input, and the pixel gray values are subjected to GF (17) domain division diffusion. The diffusion sequences are S_1 and S_2 generated by encryption steps 7 and 8. The diffusion processes are

$$\begin{cases} P_{i,H} = C_{i,H} \div C_{i-1,H} \div S_{i,H} \\ P_{i,L} = C_{i,L} \div C_{i-1,L} \div S_{i,L} \\ P_i = (P_{i,H} \times 16 + P_{i,L}) \end{cases} \tag{34}$$

$$\begin{cases} P_{i,H} = C_{i,H} \div C_{i+1,H} \div S_{i,H} \\ P_{i,L} = C_{i,L} \div C_{i+1,L} \div S_{i,L} \\ P_i = (P_{i,H} \times 16 + P_{i,L}) \end{cases} \tag{35}$$

Step 2: Scrambling recovery of the pixel matrix. Firstly, performing the zigzag inverse algorithm on the matrix, and then the matrix is divided into the block. The size of the block is identical with in the encryption step 5, and matrix blocks are reversely scrambled according to Fig. 8, and matrices after the block scrambled are combined into a large image matrix.

Step 3: The twice OMP algorithm is used to reconstruct the pixel matrix to obtain the sparse matrix before compression.

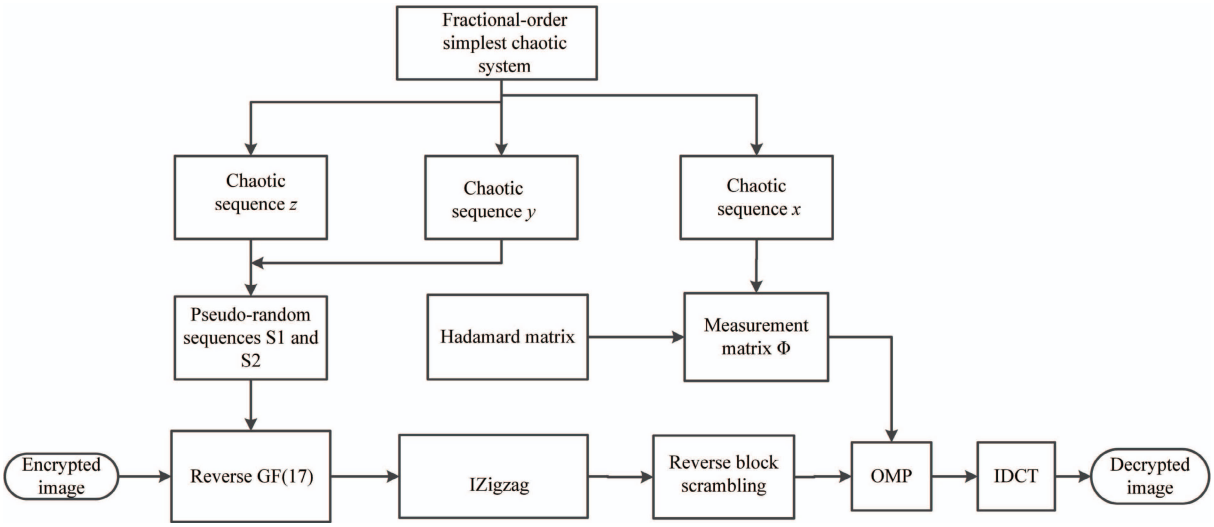


FIGURE 9. Decryption process flow chart.

Step 4: The decrypted image is obtained by performing an inverse discrete cosine transform (IDCT) algorithm on the sparse pixel matrix.

IV. SIMULATION RESULTS AND PERFORMANCE ANALYSIS

This simulation was implemented on Matlab 2018a. The workstation with Intel Core i7-6500U CPU @ 3.1 GHz, 4.00 GB memories, and operating system for Microsoft Windows 10. Setting the parameters, $\alpha = 0.6$, $\beta = 1.9$, $a = 1$, $b = 1/3$, $q = 0.45$, $h = 0.01$, initial values $[x_0, y_0, z_0] = [0.1, 0, 0]$, compression ratio $CR = 0.75$, input "Lena", "pepper", "man", "baboon", "fruits", "Tiffany", "Zelda", "house", eight 256×256 grayscale images to test this algorithm. The results are shown in Fig. 10. Obviously, the encrypted image is smaller than the original image, and the encrypted image does not recognize any plaintext image information, which means that the encryption algorithm can compress the original image and effectively encrypt the image information. Comparing the decrypted image with the original image, it can be found that the two are almost identical. It represents that the decryption algorithm can effectively reconstruct and decrypt images.

A. THE EFFECT OF COMPRESSION RATIO ON SIMULATION RESULTS

1) PEAK SIGNAL TO NOISE RATIO (PSNR)

The PSNR is usually used to assess the performance of image reconstruction. The larger the PSNR value, the more similar the image is to the original image, which is defined as follows:

$$\begin{cases} MSE = \dfrac{1}{H \times W} \displaystyle\sum_{i=1}^{H} \sum_{j=1}^{W} (F(i,j) - f(i,j))^2 \\ PSNR = 10 \cdot \log_{10}(\dfrac{255^2}{MSE}) \end{cases} \tag{36}$$

TABLE 2. PSNR values of different images under different CRs.

CR	0.25	0.5	0.75	0.9
Lena (256×256)	30.7689	33.3791	32.2754	34.9429
pepper(256×256)	30.5564	33.7346	32.4140	35.3235
man (256×256)	29.8442	32.2468	31.2159	33.5839

TABLE 3. PSNR values of Lena (256 × 256) with different algorithms.

CR	Ours	Ref. [23]	Ref. [24]	Ref. [25]	Ref. [32]
0.75	32.28	29.56	32.22	29.22	30.82
0.5	33.38	29.82	29.85	29.23	26.87
0.25	30.77	26.06	28.09	26.52	22.62

where $F(i, j)$ is the decrypted image and $f(i, j)$ is the original image. H and W represent the length and width of the image. Table 2 displays the PSNR values of different images under different CRs. In Table 3, the PSNR values of the Lena image under different CRs are compared with other algorithms. From the table, we can see that the quality of our image reconstruction is higher than other algorithms under the same image compression ratios. When the sampled data of the image is small, the proposed algorithm still obtains good reconstructed image quality.

2) MEAN STRUCTURAL SIMILARITY (MSSIM)

The MSSIM is an indicator that measures the degree of similarity between two images from three levels: brightness, contrast, and structure. The calculation process is

$$\text{MSSIM}(X, Y) = \frac{1}{M} \sum_{k=1}^{M} \text{SSIM}(X_k, Y_k) \tag{37}$$

$$\text{SSIM}(X, Y) = l(X, Y) \cdot c(X, Y) \cdot s(X, Y) \tag{38}$$

$$l(X, Y) = \frac{2u_X u_Y + C_1}{u_X^2 + u_Y^2 + C_1} \tag{39}$$

FIGURE 10. Simulation results: (a1) plaintext "Lena" (a2) Encrypted "Lena" (a3) Decrypted "Lena" (b1) plaintext "pepper" (b2) Encrypted "pepper" (b3) Decrypted "pepper" (c1) plaintext "man" (c2) Encrypted "man" (c3) Decrypted "man" (d1) plaintext "baboon" (d2) Encrypted "baboon" (d3) Decrypted "baboon" (e1) plaintext "fruits" (e2) Encrypted "fruits" (e3) Decrypted "fruits" (f1) plaintext image "Tiffany" (f2) Encrypted "Tiffany" (f3) Decrypted "Tiffany" (g1) plaintext image "Zelda" (g2) Encrypted "Zelda" (g3) Decrypted "Zelda" (h1) plaintext "house" (h2) Encrypted "house" (h3) Decrypted "house".

TABLE 4. MSSIM values of different images under different CRs.

CR	0.25	0.5	0.75	0.9
Lena (256×256)	0.6211	0.7899	0.6860	0.7988
pepper (256×256)	0.6354	0.8051	0.7146	0.8180
man (256×256)	0.5375	0.7551	0.6616	0.7830

$$c(X, Y) = \frac{2\sigma_X\sigma_Y + C_2}{\sigma_X^2 + \sigma_Y^2 + C_2} \tag{40}$$

$$s(X, Y) = \frac{\sigma_{XY} + C_3}{\sigma_X\sigma_Y + C_3} \tag{41}$$

where X is the plaintext image, Y is the decrypted image. u_X is the mean of X, u_Y is the mean of Y, σ_X^2, σ_Y^2 and σ_{XY} represent the variance and covariance of the images X and Y separately. $C_1 = (K_1 \times L)^2$, $C_2 = (K_2 \times L)^2$, $C_3 = C_2/2$ are three constants, here $K_1 = 0.01$, $K_2 = 0.03$ are the default values, and $L = 255$. The range of MSSIM values is 0-1. The larger the MSSIM values, the more similar the two images are. Table 4 indicates the MSSIM values of different images under different CRs. It can be found that the value of MSSIM changes with the change of CRs, and the quality of the image reconstructed by this algorithm is well.

TABLE 5. Key space of different encryption schemes.

Ours	Ref. [24]	Ref. [25]	Ref. [26]	Ref. [31]	Ref. [51]	Ref. [52]
2^{449}	2^{298}	2^{299}	2^{276}	2^{187}	2^{400}	2^{319}

B. KEY SPACE ANALYSIS

For an effective image cryptosystem, its key space should be large enough to withstand brute-force attacks. The key of the studied encryption algorithm is composed of chaotic system parameters a, b, α, β, h, derivative order q, initial values x_0, y_0, z_0, and the number of iterations m and n. It is assumed that the calculation accuracy of key is 10^{-15}, the key space is $(10^{15})^9 = 10^{135} \approx 2^{449}$, so the key space of the encryption algorithm proposed can effectively prevent brute-force attacks. The key space in this paper is compared with other methods in Table 5.

C. KEY SENSITIVITY ANALYSIS

To test the key sensitivity of the proposed algorithm, the key are changed 10^{-15}, and the Lena image is used as the key sensitivity test image. Fig. 11 illustrates the decrypted Lena image after the key changed 10^{-15}. Obviously, even though

TABLE 6. The variance values of histograms of the cipher images using different keys.

Images	K_0	a	b	α	β	h	q	x_0	y_0	z_0
Lena	5489	5479	5422	5474	5466	5491	5437	5439	5501	5465
Pepper	5445	5426	5437	5472	5456	5502	5456	5459	5475	5449
Man	5479	5484	5494	5473	5461	5448	5466	5489	5450	5457
Baboon	5490	5439	5481	5455	5470	5447	5421	5433	5449	5491

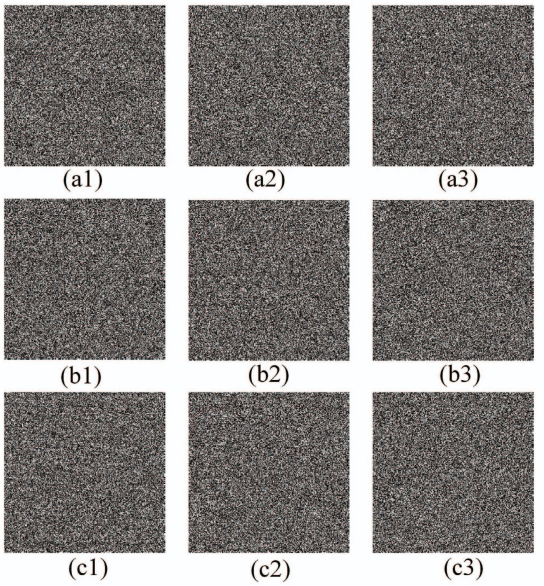


FIGURE 11. Key sensitivity test results (a1) $a + 10^{-15}$ (a2) $b + 10^{-15}$ (a3) $\alpha + 10^{-15}$ (b1) $\beta + 10^{-15}$ (b2) $h + 10^{-15}$ (b3) $q + 10^{-15}$ (c1) $x_0 + 10^{-15}$ (c2) $y_0 + 10^{-15}$ (c3) $z_0 + 10^{-15}$.

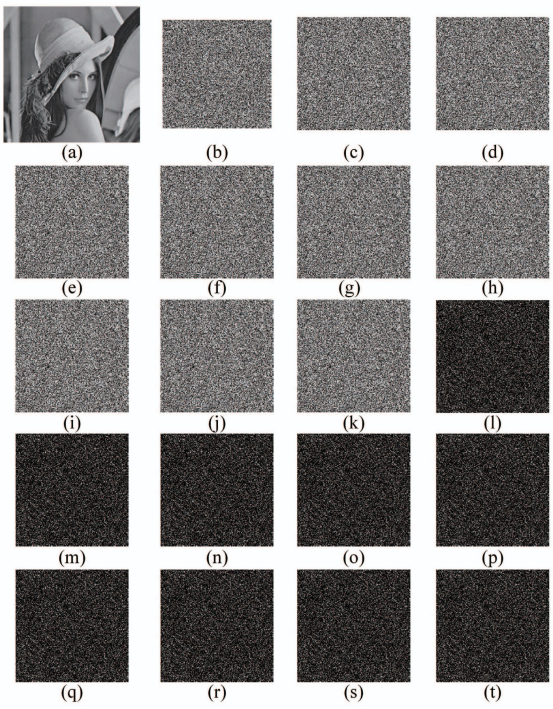


FIGURE 12. Key sensitivity test results: (a) Original Lena image; (b) Original cipher image C; (c) Encrypted Lena image using $a + 10^{-15}$; (d) Encrypted Lena image using $b + 10^{-15}$; (e) Encrypted Lena image using $\alpha + 10^{-15}$; (f) Encrypted Lena image using $\beta + 10^{-15}$; (g) Encrypted Lena image using $h + 10^{-15}$; (h) Encrypted Lena image using $q + 10^{-15}$; (i) Encrypted Lena image using $x_0 + 10^{-15}$; (j) Encrypted Lena image using $y_0 + 10^{-15}$; (k) Encrypted Lena image using $z_0 + 10^{-15}$; (l) Difference between Fig. 12(b) and (c); (m) Difference between Fig. 12(b) and (d); (n) Difference between Fig. 12(b) and (e); (o) Difference between Fig. 12(b) and (f); (p) Difference between Fig. 12(b) and (g); (q) Difference between Fig. 12(b) and (h); (r) Difference between Fig. 12(b) and (i); (s) Difference between Fig. 12(b) and (j); (t) Difference between Fig. 12(b) and (k).

the key changes are very tiny, the image cannot be decrypted normally. It indicates that the secret key is sufficiently sensitive. In addition, as shown in Fig. 12, we tested the difference between the two cipher images obtained after encrypting the same Lena image when the key changes slightly. The test results show that when the key is slightly changed, the two cipher images got by encrypting the same plain image are significantly different. It also proves that the secret key of the proposed algorithm is sufficiently sensitive.

D. STATISTICAL ANALYSIS

In this section, the algorithm's resistance to statistical attacks is evaluated by analyzing the histogram and the correlation between adjacent pixels.

1) HISTOGRAM

The histogram can intuitively reflect the distribution of the pixel values of the image, so Fig. 13 draws the histogram of the plaintext image and the cipher image. When in the range of 0 to 255, the histogram pixel values of the original image are unevenly distributed, but the encrypted image pixel values are uniformly distributed, and the histograms of different images after encryption are nearly the same. Obviously, the statistical characteristics of plaintext images have undergone fundamental changes, so the effective information of the image cannot be obtained through statistical attacks.

In addition, we use the variance of the histogram to evaluate the uniformity of the pixel distribution of the cipher image. The smaller the calculated variance value, the better the uniformity of the cipher image. Table 6 lists the variance of the four cipher images of Lena, pepper, man, and baboon. The second column of Table 6 is the variance values of the cipher image under the original secret key K_0. The other columns are the variance values of the cipher image when only one key is changed. Table 6 shows that the average variance of the cipher images are about 5450, while the variance values of the plain images are about 621874, so the

TABLE 7. The variance values of histograms of the cipher images using different keys.

Images	$a(\%)$	$b(\%)$	$\alpha(\%)$	$\beta(\%)$	$h(\%)$	$q(\%)$	$x_0(\%)$	$y_0(\%)$	$z_0(\%)$
Lena	0.1	1.2	1.3	0.4	0.03	0.9	0.9	0.2	0.4
Pepper	0.3	0.1	0.4	0.2	1.0	0.2	0.3	0.6	0.07
Man	0.09	0.3	0.1	0.3	0.6	0.2	0.2	0.5	0.4
Baboon	0.9	0.2	0.6	0.4	0.8	1.3	1.0	0.7	0.01

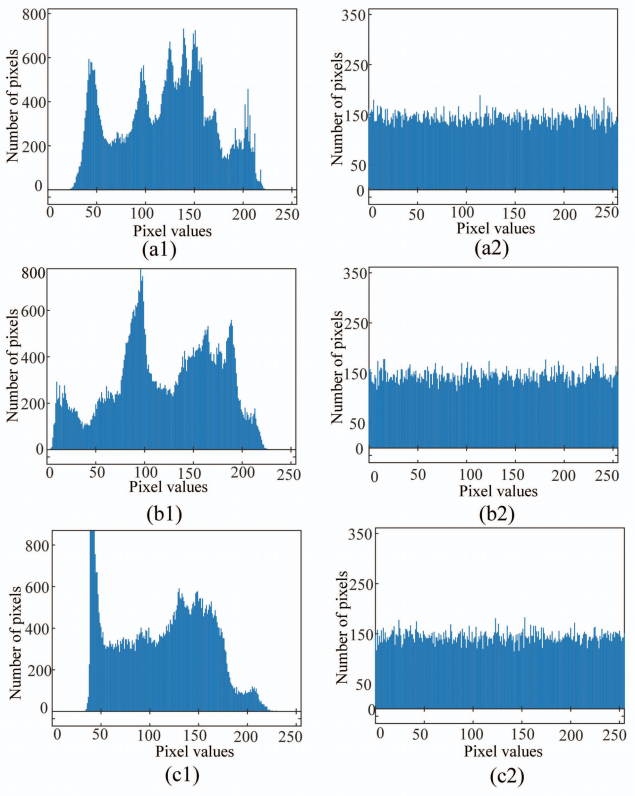


FIGURE 13. Histogram of the plaintext and cipher image (a1) Plaintext "Lena" (a2) Cipher "Lena" (b1) Plaintext "Pepper" (b2) Cipher "Pepper" (c1) Plaintext "Man" (c2) Cipher "Man".

proposed encryption scheme is effective. We study the impact of changing the key on the consistency of the cipher image by calculating the percentage of the variance of the two cipher images. The calculation results are listed in Table 7. As shown in Table 7, we found that the variance of different images fluctuates differently, and the average variance fluctuates very little, with the largest fluctuation being only 1.3%. Moreover, compared with the encryption scheme [51], the average variance fluctuates much smaller. The fluctuation value of the key a for the Lena image is 1.2%, but the fluctuation value for the pepper image is only 0.1%. It means that the histogram of proposed algorithm is sensitive to plain images. Therefore, any statistical attack is invalid for this scheme.

2) CORRELATION BETWEEN ADJACENT PIXELS

Generally, an unencrypted ordinary image has a high correlation between adjacent pixels, especially in the horizontal, vertical, and diagonal directions. The encrypted image has almost no correlation between its adjacent pixels.

The calculation formula for the correlation coefficient is

$$\rho_{uv} = \frac{\mathrm{cov}(u, v)}{\sqrt{D(u)}\sqrt{D(v)}} \tag{42}$$

$$\mathrm{cov}(u, v) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(u))(y_i - E(v)) \tag{43}$$

$$D(u) = \frac{1}{N}\sum_{i=1}^{N}(u_i - E(u))^2 \tag{44}$$

$$E(u) = \frac{1}{N}\sum_{i=1}^{N}u_i \tag{45}$$

where $E(u)$ and $E(v)$ are the mean of adjacent pixels u and v, respectively, $D(u)$ and $D(v)$ are the variance of u and v, and N is the number of all pixels in the image.

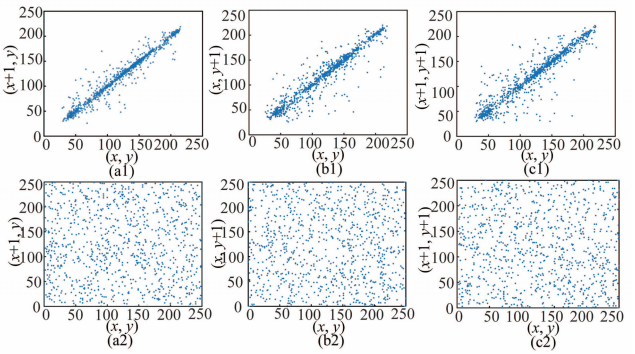


FIGURE 14. Pixel value distribution of original image and cipher image (a1) Horizontal direction of Lena image, (a2) Horizontal direction of cipher Lena image, (b1) Vertical direction of Lena image, (b2) Vertical direction of cipher Lena image, (c) Diagonal direction of Lena image, (c2) Diagonal direction of cipher Lena image.

TABLE 8. Correlation coefficients of adjacent pixels.

	Images	Horizontal	Vertical	Diagonal
Lena	Original image	0.9720	0.9464	0.9223
	Encrypted image	-0.0046	-0.0002	0.0005
Pepper	Original image	0.9714	0.9654	0.9395
	Encrypted image	0.0003	0.0011	-0.0030
Man	Original image	0.9570	0.9435	0.9130
	Encrypted image	-0.0048	0.0018	-0.0022

The distribution of adjacent pixels of the original Lena image and the cipher Lena image in different directions is shown in Fig. 14. From the figure, the adjacent pixels of the plaintext image of various directions are distributed among the diagonal of the coordinate axis, whereas the adjacent pixels of the encrypted image are evenly distributed over 0-250. The correlation coefficients of adjacent pixels in various directions in different images are given in Table 8.

TABLE 9. Correlation coefficients of adjacent pixels in different algorithms.

Direction	Lean	Ours	Ref. [22]	Ref. [33]	Ref. [53]	Ref. [54]	Ref. [10]	Ref. [52]
Horizontal	0.9720	-0.0046	0.0062	0.0104	0.0020	0.0024	0.0008	0.0019
Vertical	0.9465	-0.0002	-0.0107	0.0299	0.0007	-0.0006	0.0008	0.0038
Diagonal	0.9223	0.0005	0.0052	0.0062	0.0014	0.0012	0.0008	-0.0019

From Table 8, the correlation coefficients between adjacent pixels of plaintext images in various directions is very high, and the correlation coefficients of encrypted images approach 0, indicating that the proposed algorithm can effectively break the correlation between pixels. Table 9 compares the correlation coefficients of Lena images in different literatures and the algorithm.

E. INFORMATION ENTROPY

The information entropy is an indicator that reflects the uncertainty of image information distribution, the stronger the randomness, the larger the entropy value. It can be expressed as

$$H(S) = -\sum_{i=0}^{N-1} P(S_i)\log_2[P(S_i)] \tag{46}$$

here N indicates that S_i has N different values, and $P(S_i)$ means the probability that S_i appears in the image S. For a plaintext image of $S = 256$, the theoretical value $H(S) = 8$. The entropy of different images before and after encryption is shown in Table 10. Table 11 compares the information entropy of encrypted Lena image with different method. From Tables 10 and 11, information entropy of the encryption algorithm is close to 8, so it has strong randomness and can resist statistical attacks well.

TABLE 10. Information entropy of different images.

	Lena	pepper	man
Plaintext image	7.4127	7.5570	7.2283
Cipher image	7.9951	7.9954	7.9948

F. ROBUSTNESS ANALYSI

The robustness is an important evaluation criterion for cryptographic systems. The encrypted images may lose data during transmission and processing. Therefore, a good cryptosystem must have strong robustness to withstand data loss.

To assess the robustness of the proposed algorithm, three data loss with different amounts was generated for the encrypted Lena image. The encrypted Lena images after different data loss are shown in Fig. 15(a1), (b1) and (c1), and decrypted images are shown in Fig. 15(a2), (b2) and (c2). It shows that although the data of the encrypted image is lost, the decrypted image still contains the main information of the original image. It explains that the studied algorithm has strong robust performance.

G. KNOWN/CHOSEN PLAINTEXT ATTACKS ANALYSIS

An effective cryptographic system should be able to resist four typical attacks, such as ciphertext only, known plaintext, chosen ciphertext, chosen plaintext, etc. Among them, the chosen plaintext attack is the most powerful. If the cryptographic system is capable of such an attack, it can also resist several other attacks [60]. In this part, we use known/chosen plaintext attacks to evaluate the security of encryption algorithms. Hackers usually choose a random matrix to obtain the corresponding cipher and guess the key structure. And a completely black or completely white image will invalidate the encryption scrambling algorithm. In order to measure the performance of the algorithm against known plaintext attacks and selected plaintext attacks, we use all white and all black pictures as the encryption objects. The encryption effect is shown in Fig. 16.

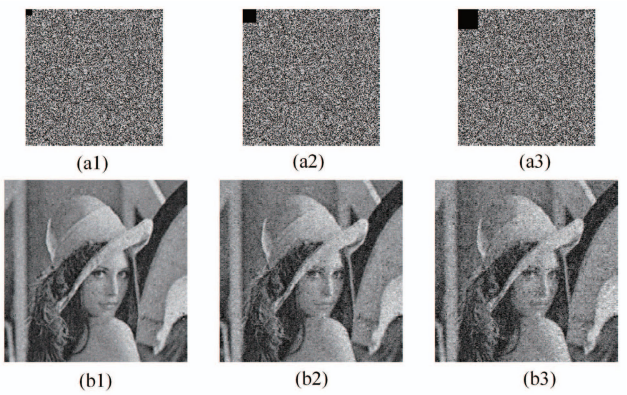


FIGURE 15. Robustness analysis results (a1) The encrypted image with 5% data loss (a2) The decrypted image with 5% data loss (b1) The encrypted image with 10% data loss (b2)The decrypted image with 10% data loss (c1) The encrypted image with 15% data loss (c2) The decrypted image with 15% data loss.

From the Fig. 16, the encrypted image cannot identify any information of the original image, and the pixel values of the encrypted image are evenly distributed. In addition, it can be seen from Table 12 that the information entropy, NPCR and UACI are close to theoretical values, the correlation coefficient of the ciphertext image is close to 0. Therefore, the algorithm can resist known/chosen plaintext attacks. Moreover, compared with schemes [61] and [62], the calculated data is closer to the theoretical value, so the proposed algorithm has higher security.

H. RANDOMNESS OF CIPHER IMAGES

In order to resist statistical attacks, the pixels of an ideal cipher image need to be evenly distributed. We use NIST SP 800-22 to measure whether the pixels of the ciphertext image are evenly distributed. When P-value>0.0001, the measurement sequence is randomly and uniformly distributed.

TABLE 11. Information entropy of different algorithms.

Algorithm	Ours	Ref. [25]	Ref. [55]	Ref. [56]	Ref. [57]	Ref. [58]	Ref. [59]
Lena	7.9951	7.9935	7.9826	7.9404	7.9941	7.9943	7.9872

TABLE 12. The performance analysis of all-white and all-black images.

Image(256×256)	Information entropy	UACI(%)	NPCR(%)	Correlation coefficients		
				Horizontal	Vertical	Diagonal
All-white	7.9951	33.4891	99.6063	-0.0082	-0.0071	0.0015
All-black	7.9957	33.4853	99.6080	-0.0005	-0.0067	-0.0011

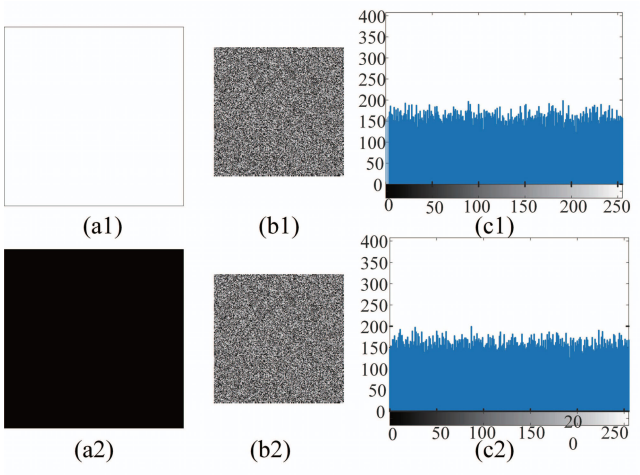


FIGURE 16. The analysis result. (a) Original image. (b) Encrypted image. (c) Histogram of cipher image.

TABLE 13. The randomness test result of the cipher images.

Test Name	P-value	Pass rate	
Frequency	0.534146	0.99	pass
Block Frequency	0.145326	1	pass
Cumulative Sums	0.463393	1	pass
Runs	0.983453	0.99	pass
Longest Run	0.574903	0.99	pass
Rank	0.162606	0.98	pass
FFT	0.115387	0.98	pass
Non Overlapping Template	0.492798	0.99	pass
Overlapping Template	0.137282	1	pass
Universal	0.534146	0.99	pass
Approximate Entropy	0.350485	1	pass
Random Excursions	0.4007377	0.99	pass
Random Excursions Variant	0.482630	0.99	pass
Serial	0.668026	0.97	pass
Linear Complexity	0.955835	0.99	pass

We choose 22 images with a size of 1024 × 1024, set the compression rate is 0.75, and the size of the encrypted image is 768 × 768, and then converts each pixel of the cipher images into an eight-bit binary sequence. The total length of the obtained sequence is 22 × 768 × 768 × 8.103809024. Table 13 lists the test results. It can be seen that the P-values are all greater than 0.0001, which indicates that the cipher images have very high randomness.

TABLE 14. Encryption speed test results.

Schemes	Our	Ref. [63]	Ref. [53]	Ref. [64]	Ref. [65]
Time(s)	7.07	5.89	7.53	4.98	8.86

I. TIME ANALYSIS

The execution time of the encryption process is also an important measure of the encryption scheme. We analyze the encryption time of the Lena image and compare it with the previous algorithm [53], [63]–[65] as shown in Table 14. Table 14 shows that the encryption time of the proposed algorithm is faster than that of schemes [53], [65], but obviously slower than that of schemes [63], [64]. The main reason is that to improve the security of the encryption algorithm, the chaotic system runs longer when performing complex diffusion algorithms on images. In short, in order to ensure a safe and effective encryption effect, the encryption time consumption is large, which is lower than some other algorithms.

V. CONCLUSION

A fractional-order simplest memristive chaotic circuit is established. The dynamic analysis indicates that the fractional-order simplest memristive chaotic system is highly sensitive to initial values and parameters. The chaotic sequences generated by the system have good randomness. Therefore, it can be better applied to digital image password system. Based on the fractional-order simplest memristive chaotic system and compressive sensing theory, a digital image encryption algorithm is proposed. The compressing images of CS theory are used to decrease the costs of image transmission and storage, and improve reconstruction accuracy. Simulation results display that the algorithm has good compression and reconstruction performance. Even the $CR = 0.25$, the obtained PSNR values and MSSIM values are large enough to still identify the main information of the original image. The analysis of security performance indicates that the algorithm has the ability to effectively prevent the various attacks, such as statistical attack, robustness attack and known/chosen plaintext attacks. In addition, the key space of 2^{449} is much larger than 2^{100}, which can prevent brute force attacks, and the algorithm security is high. Therefore, it has a good application prospect in the field of image secure transmission.

REFERENCES

[1] C. Cao, K. Sun, and W. Liu, "A novel bit-level image encryption algorithm based on 2D-LICM hyperchaotic map," *Signal Process.*, vol. 143, pp. 122–133, Feb. 2018.

[2] C. Li, G. Luo, K. Qin, and C. Li, "An image encryption scheme based on chaotic tent map," *Nonlinear Dyn.*, vol. 87, no. 1, pp. 127–133, Jan. 2017.

[3] Z. Hua, Y. Zhou, C.-M. Pun, and C. L. P. Chen, "2D sine logistic modulation map for image encryption," *Inf. Sci.*, vol. 297, pp. 80–94, Mar. 2015.

[4] X.-Y. Wang, S.-X. Gu, and Y.-Q. Zhang, "Novel image encryption algorithm based on cycle shift and chaotic system," *Opt. Lasers Eng.*, vol. 68, pp. 126–134, May 2015.

[5] Q. Zhang, L. Guo, and X. Wei, "Image encryption using DNA addition combining with chaotic maps," *Math. Comput. Model.*, vol. 52, nos. 11–12, pp. 2028–2035, Dec. 2010.

[6] W. Liu, K. Sun, and C. Zhu, "A fast image encryption algorithm based on chaotic map," *Opt. Lasers Eng.*, vol. 84, pp. 26–36, Sep. 2016.

[7] X.-Y. Wang, Y.-Q. Zhang, and Y.-Y. Zhao, "A novel image encryption scheme based on 2-D logistic map and DNA sequence operations," *Nonlinear Dyn.*, vol. 82, no. 3, pp. 1269–1280, Nov. 2015.

[8] L.-M. Zhang, K.-H. Sun, W.-H. Liu, and S.-B. He, "A novel color image encryption scheme using fractional-order hyperchaotic system and DNA sequence operations," *Chin. Phys. B*, vol. 26, no. 10, Sep. 2017, Art. no. 100504.

[9] X. Chai, Y. Chen, and L. Broyde, "A novel chaos-based image encryption algorithm using DNA sequence operations," *Opt. Lasers Eng.*, vol. 88, pp. 197–213, Jan. 2017.

[10] X. Wang and S. Gao, "Image encryption algorithm based on the matrix semi-tensor product with a compound secret key produced by a Boolean network," *Inf. Sci.*, vol. 539, pp. 195–214, Oct. 2020.

[11] X. Wang, L. Feng, and H. Zhao, "Fast image encryption algorithm based on parallel computing system," *Inf. Sci.*, vol. 486, pp. 340–358, Jun. 2019.

[12] X. Wang and S. Gao, "A chaotic image encryption algorithm based on a counting system and the semi-tensor product," *Multimedia Tools Appl.*, pp. 1–22, Nov. 2020, doi: 10.1007/s11042-020-10101-6.

[13] S. Li, B. Zhang, S. Zhao, and J. Yang, "Local discriminant coding based convolutional feature representation for multimodal finger recognition," *Inf. Sci.*, vol. 547, pp. 1170–1181, Feb. 2021.

[14] H. Liu and X. Wang, "Color image encryption using spatial bit-level permutation and high-dimension chaotic system," *Opt. Commun.*, vol. 284, nos. 16–17, pp. 3895–3903, Aug. 2011.

[15] H. Liu and X. Wang, "Color image encryption based on one-time keys and robust chaotic maps," *Comput. Math. Appl.*, vol. 59, no. 10, pp. 3320–3327, May 2010.

[16] X.-Y. Wang, L. Yang, R. Liu, and A. Kadir, "A chaotic image encryption algorithm based on perceptron model," *Nonlinear Dyn.*, vol. 62, no. 3, pp. 615–621, Nov. 2010.

[17] 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.

[18] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[19] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[20] Y. Rachlin and D. Baron, "The secrecy of compressed sensing measurements," in *Proc. 46th Annu. Allerton Conf. Commun., Control, Comput.*, Sep. 2008, pp. 813–817.

[21] M. Ramezani Mayiami, B. Seyfe, and H. G. Bafghi, "Perfect secrecy via compressed sensing," in *Proc. Iran Workshop Commun. Inf. Theory*, May 2013, pp. 1–5.

[22] H. Wang, D. Xiao, M. Li, Y. Xiang, and X. Li, "A visually secure image encryption scheme based on parallel compressive sensing," *Signal Process.*, vol. 155, pp. 218–232, Feb. 2019.

[23] 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.

[24] J. Mou, F. Yang, R. Chu, and Y. Cao, "Image compression and encryption algorithm based on hyper-chaotic map," *Mobile Netw. Appl.*, pp. 1–13, Jun. 2019, doi: 10.1007/s11036-019-01293-9.

[25] Q. Xu, K. Sun, C. Cao, and C. Zhu, "A fast image encryption algorithm based on compressive sensing and hyperchaotic map," *Opt. Lasers Eng.*, vol. 121, pp. 203–214, Oct. 2019.

[26] 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.

[27] X. Wang and D. Xu, "A novel image encryption scheme based on Brownian motion and PWLCM chaotic system," *Nonlinear Dyn.*, vol. 75, nos. 1–2, pp. 345–353, Jan. 2014.

[28] X. Chai, Z. Gan, Y. Chen, and Y. Zhang, "A visually secure image encryption scheme based on compressive sensing," *Signal Process.*, vol. 134, pp. 35–51, May 2017.

[29] R. Huang, K. H. Rhee, and S. Uchida, "A parallel image encryption method based on compressive sensing," *Multimedia Tools Appl.*, vol. 72, no. 1, pp. 71–93, Sep. 2014.

[30] Y. Luo, R. Zhou, J. Liu, Y. Cao, and X. Ding, "A parallel image encryption algorithm based on the piecewise linear chaotic map and hyper-chaotic map," *Nonlinear Dyn.*, vol. 93, no. 3, pp. 1165–1181, Aug. 2018.

[31] 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.

[32] N. Zhou, A. Zhang, J. Wu, D. Pei, and Y. Yang, "Novel hybrid image compression–encryption algorithm based on compressive sensing," *Optik*, vol. 125, no. 18, pp. 5075–5080, 2014.

[33] N. Zhou, H. Li, D. Wang, S. Pan, and Z. Zhou, "Image compression and encryption scheme based on 2D compressive sensing and fractional Mellin transform," *Opt. Commun.*, vol. 343, pp. 10–21, May 2015.

[34] X. Chai, J. Bi, Z. Gan, X. Liu, Y. Zhang, and Y. Chen, "Color image compression and encryption scheme based on compressive sensing and double random encryption strategy," *Signal Process.*, vol. 176, Nov. 2020, Art. no. 107684.

[35] A. Kiani-B, K. Fallahi, N. Pariz, and H. Leung, "A chaotic secure communication scheme using fractional chaotic systems based on an extended fractional Kalman filter," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 14, no. 3, pp. 863–879, Mar. 2009.

[36] S. Bhalekar, "Dynamical analysis of fractional order Uçar prototype delayed system," *Signal, Image Video Process.*, vol. 6, no. 3, pp. 513–519, Sep. 2012.

[37] 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. e0119660.

[38] 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.

[39] 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, pp. 1–12, Jan. 2018.

[40] 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, Jun. 2016.

[41] 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.

[42] J. Zhao, S. Wang, Y. Chang, and X. Li, "A novel image encryption scheme based on an improper fractional-order chaotic system," *Nonlinear Dyn.*, vol. 80, no. 4, pp. 1721–1729, Jun. 2015.

[43] X. Huang, T. Sun, Y. Li, and J. Liang, "A color image encryption algorithm based on a fractional-order hyperchaotic system," *Entropy*, vol. 17, no. 1, pp. 28–38, Dec. 2014.

[44] Z. Wang, X. Huang, N. Li, and X.-N. Song, "Image encryption based on a delayed fractional-order chaotic logistic system," *Chin. Phys. B*, vol. 21, no. 5, May 2012, Art. no. 050506.

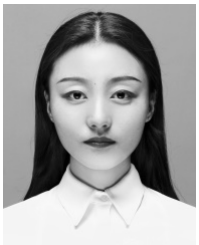
[45] M. Ahmad, I. R. Khan, and S. Alam, "Cryptanalysis of image encryption algorithm based on fractional-order lorenz-like chaotic system," in *Proc. Emerging ICT Bridging Future 49th Annu. Conv. Comput. Soc. India CSI*, vol. 338, 2015, pp. 381–388.

[46] X. Wu, "A color image encryption algorithm using the fractional-order hyperchaotic systems," in *Proc. Int. Workshop Chaos-Fractals Theories Appl.*, 2012, pp. 196-201.

[47] Y. Xu, H. Wang, Y. Li, and B. Pei, "Image encryption based on synchronization of fractional chaotic systems," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 19, no. 10, pp. 3735–3744, Oct. 2014.

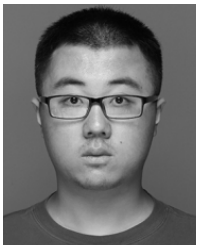
[48] Y. Wang and S. Zhou, "Image encryption algorithm based on fractional-order Chen chaotic system," *J. Comput. Appl.*, vol. 33, no. 4, pp. 1043–1046, Oct. 2013.

[49] 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.
[50] F. Yang, J. Mou, C. Ma, and Y. Cao, "Dynamic analysis of an improper fractional-order laser chaotic system and its image encryption application," *Opt. Lasers Eng.*, vol. 129, Jun. 2020, Art. no. 106031.
[51] 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.
[52] X. Wang, L. Liu, and Y. Zhang, "A novel chaotic block image encryption algorithm based on dynamic random growth technique," *Opt. Lasers Eng.*, vol. 66, pp. 10–18, Mar. 2015.
[53] 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.
[54] 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.
[55] Y. Luo, M. Du, and J. Liu, "A symmetrical image encryption scheme in wavelet and time domain," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 20, no. 2, pp. 447–460, Feb. 2015.
[56] L. Guo, J. Chen, and J. Li, "Chaos-based color image encryption and compression scheme using DNA complementary rule and Chinese remainder theorem," in *Proc. 13th Int. Comput. Conf. Wavelet Act. Media Technol. Inf. Process. (ICCWAMTIP)*, Dec. 2016, pp. 208–212.
[57] X. Hu, L. Wei, W. Chen, Q. Chen, and Y. Guo, "Color image encryption algorithm based on dynamic chaos and matrix convolution," *IEEE Access*, vol. 8, pp. 12452–12466, 2020.
[58] F. Musanna and S. Kumar, "A novel fractional order chaos-based image encryption using Fisher Yates algorithm and 3-D cat map," *Multimedia Tools Appl.*, vol. 78, no. 11, pp. 14867–14895, Jun. 2019.
[59] M. Kar, M. K. Mandal, D. Nandi, A. Kumar, and S. Banik, "Bit-plane encrypted image cryptosystem using chaotic, quadratic, and cubic maps," *IETE Tech. Rev.*, vol. 33, no. 6, pp. 651–661, Nov. 2016.
[60] X. Wang, L. Teng, and X. Qin, "A novel colour image encryption algorithm based on chaos," *Signal Process.*, vol. 92, no. 4, pp. 1101–1108, Apr. 2012.
[61] N. Chidambaram, P. Raj, K. Thenmozhi, and R. Amirtharajan, "Advanced framework for highly secure and cloud-based storage of colour images," *IET Image Process.*, vol. 14, no. 13, pp. 3143–3153, Nov. 2020.
[62] R. Sivaraman, S. Rajagopalan, J. B. B. Rayappan, and R. Amirtharajan, "Ring oscillator as confusion–diffusion agent: A complete TRNG drove image security," *IET Image Process.*, vol. 14, no. 13, pp. 2987–2997, Nov. 2020.
[63] X. Chai, H. Wu, Z. Gan, Y. Zhang, and Y. Chen, "Hiding cipher-images generated by 2-D compressive sensing with a multi-embedding strategy," *Signal Process.*, vol. 171, Jun. 2020, Art. no. 107525.
[64] L. Xu, Z. Li, J. Li, and W. Hua, "A novel bit-level image encryption algorithm based on chaotic maps," *Opt. Lasers Eng.*, vol. 78, pp. 17–25, Mar. 2016.
[65] S. Amina and F. K. Mohamed, "An efficient and secure chaotic cipher algorithm for image content preservation," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 60, pp. 12–32, Jul. 2018.

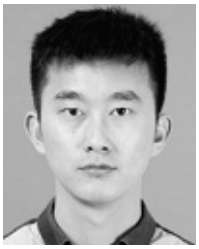


HAIYING HU received the B.S. degree from Dalian Polytechnic University, Dalian, China, in 2019, where she is currently pursuing the Ph.D. degree in control science and engineering. Her main research interests include chaos theory and chaotic digital image cryptosystem.

YINGHONG CAO received the B.S. degree in electronic engineering and the Ph.D. degree in signal and information processing from the Dalian University of Technology (DUT), Dalian, China, in 2003 and 2013, respectively. She is currently a Lecturer with the School of Information Science and Engineering, Dalian Polytechnic University. Her research interests include communication signal processing, speech processing, and the Internet of Things Technology and Application.



JI XU received the B.E. degree from the Shengli College China University of Petroleum, China, in 2017. He is currently pursuing the Ph.D. degree with Dalian Polytechnic University, Dalian, China. His main research interests include chaos theory and chaotic digital image cryptosystem.



CHENGUANG MA received the B.S. degree from Tianjin Agricultural University, Tianjin, China, in 2017. He is currently pursuing the Ph.D. degree with Dalian Polytechnic University, Dalian, China. His main research interest includes chaos theory and application.

HUIZHEN YAN received the B.S. and M.S. degrees from Xi'an Jiaotong University (XJU), Xi'an, China, and the Ph.D. degree in applied mathematics from Northeastern University (NEU), Shenyang, China, in 2000. She is currently a Professor with the School of Information Science and Engineering, Dalian Polytechnic University. Her research interests include game theory and its application, and ecological mathematics.