IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 22, 2021, accepted June 29, 2021, date of publication July 5, 2021, date of current version July 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3094563

An Image Encryption Scheme Based on IAVL Permutation Scheme and DNA Operations

YUWEN SHA, YINGHONG CAO, HUIZHEN YAN, XINYU GAO, AND JUN MOU
School of Information Science and Engineering, Dalian Polytechnic University, Dalian 116034, China

Corresponding authors: Yinghong Cao (caoyinghong@dlpu.edu.cn) and Jun Mou (moujun@csu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 62061014, in part by the Natural Science Foundation of Liaoning Province under Grant 2020-MS-274, and in part by the Basic Scientific Research Projects of Colleges and Universities of Liaoning Province under Grant J202148.

ABSTRACT In this paper, a novel image encryption algorithm based on a new permutation scheme and DNA operations are introduced. In our algorithm, SHA 256 and DNA hamming distance participate in the generation of the initial conditions of the 4D chaotic system, which can associate the encryption system with the original image. In the permutation process, based on the adjustment process of the IAVL (improved balanced binary tree), a new scrambling algorithm is constructed. Then the dynamic block coding rules are designed, in which different image blocks have different coding rules. In the diffusion process, a new diffusion algorithm with intra-block and inter-block is proposed to perform DNA operations on the intermediate encryption result and the key matrix. In the security analysis, the key space of the encryption system is 2^{933} and the information entropy is about 7.9973. In addition, the NPCR and UACI in the differential attack test are close to the ideal values of 99.6094% and 33.4653%. To further prove the security of the encryption algorithm, the Irregular deviation, Maximum deviation, Energy, Contrast, and Homogeneity tests are introduced into the analysis. Experimental results illustrate that the encryption scheme can against multiple illegal attacks like statistical, brute-force and differential attacks.

INDEX TERMS Image encryption, 4D chaotic system, DNA operations.

I. INTRODUCTION

As an important carrier of information, image files are frequently transmitted over the public Internet. So how to prevent eavesdropping becomes a major challenge. Traditional text information has some mature encryption technologies, including RSA, DES and AES [1], but they cannot meet the needs of image encryption due to the large size and special storage format of the image files. In recent years, image encryption technology has attracted the attention of more and more scholars and experts [2]–[8]. Various encryption techniques like SCAN [9], [10], elliptic curve [11], Fourier transform [12], cellular automata [13] and wavelet transform [14] have been introduced by the research community for image encryption which based on the chaotic systems. In 1998, image encryption based on the architecture of scrambling and diffusion has been proposed by Fridrich's for image encryption [15]. Until now, many chaotic-based encryption schemes still use this framework [5], [16]–[32]. For example, a new three-dimensional chaotic system for image encryption was used in [16]. They change the plain images to a 3D cubic DNA matrix to reduce encryption time. Nepomuceno *et al.* [17] used 1D chaotic map of the difference of two pseudo-orbits to encrypt image. To increase the encryption key space, the research community finds that multiple chaotic maps are often used in combination in some image encryption. Based on using a dynamical state variables selection mechanism, a fast image encryption technique was proposed by Chen *et al.* [21]. The two-dimensional area-preserving chaotic map was used to produce the pseudo-random sequences for the permutation stage and the key matrix are obtained by a one-dimensional chaotic system for the diffusion stage. Kulsoom *et al.* [19] proposed an efficient encryption technology in which pixels of image is permuted using PWLCM system and diffusion is operated using 1D Logistic map. Zhu and Zhu [20] suggested a plaintext-related image encryption scheme using 5D chaotic map in which scrambled image is divided into lots of blocks and diffused within the block, authors stated that the scheme can resist

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

selected plaintext attacks due to permutation operation is related to the plain image. One-time pad is currently the most secure encryption algorithm, but its password storage has a high cost. Therefore, more and more researchers have turned their attention to the field of DNA encryption.

The DNA computing algorithm has advantages which include high parallelism, ultra-low power consumption and big information density. So, image encryption combined with the DNA computing algorithm and the chaotic map is born which is used to solve this problem [18], [33]–[43]. An image encryption based on dynamic DNA encoding was proposed by Aditya *et al.* [33]. In the encryption scheme, dynamic DNA encoding and pixel scrambling are applied. Patro *et al.* [34] suggested an efficient image encryption scheme, which is secure, lossless, and noise-resistive using chaos, hyper-chaos, and DNA sequence operations. A block encryption algorithm for gray images was introduced by Rehman *et al.* [35]. In the selective diffusion process, higher and lower bit planes of an image are encoding into DNA sequences, respectively and divided into blocks to implement DNA algebraic operation with each other. Finally, the higher bit planes of blocks are replaced by the result.

There are other algorithms that can help improve the proposed encryption system [2], [44]–[58]. For example, RGB DNA image encryption technology was designed in [44]. But Ozakaynak *et al.* [53] claimed that it has performance defects. Firstly, the equivalent key may be cracked by several special plain images. Second, the security is poor in the face of chosen-plaintext attack. Yang *et al.* [45] used a hyperchaotic sequence to improve the color image encryption scheme in which DNA encoding rule fixed 1 and decoding fixed 3. The fixed image encoding/decoding rules may be to expand the key space, but the security is not as good as dynamic ones. the encryption process is not sensitive enough to changes to the keys or the original images. In order to improve the disadvantages of some common encryption systems and enhance the performance of the system, a safe and efficient encryption scheme is urgently needed to be proposed.

Based on the above analysis, an encryption scheme combining the chaotic system and DNA manipulation is proposed. Our algorithm has three advantages: First, a new 4D chaotic system is designed in our encryption scheme [7]. Through dynamical analysis, it is found that this system has many control parameters and any small change in them will create a new state of chaos. So, it can use to generate a large number of pseudo-random numbers for image encryption. Second, the process of adjusting IBST to IAVL is accompanied by the exchange and movement of nodes. Therefore, a new scrambling algorithm is designed by referring to the IAVL adjustment process. Compared with the previous circular scrambling of rows and columns, the new scrambling algorithm also has column exchange operations, which greatly increases the flexibility of the scrambling algorithm. Third, during the diffusion process, the dynamic coding of DNA and the intra-block and inter-block diffusion technology are applied, which will greatly improve the diffusion effect of the image.

The paper structure is outlined as follows. In Sec. II, preliminary materials are given. In Sec. III the details of our encryption scheme are introduced. Then simulation tests are shown in Sec. IV. Finally, Sec. V and Sec. VI give security analysis and paper conclusion, respectively.

II. PRELIMINARY MATERIALS

A. THE MODEL OF THE CHAOTIC SYSTEM

In this section, a new chaotic system model will be cited. After the initial values and parameters are given, the dynamical characteristics are then analyzed to prove the applicability of this chaotic system in the encryption field.

1) 4D CHAOTIC SYSTEM

The new four-dimensional chaotic system is introduced as follows

$$\begin{cases} \dot{x} = r(\alpha + \beta u)z \\ \dot{y} = -c(\alpha + \beta u)z - dy + e(\alpha + \beta u)^2 z^2 y \\ \dot{z} = -x - (ay^2 - b)(\alpha + \beta u)z \\ \dot{u} = z \end{cases} \tag{1}$$

where x, y, z and u make up the system states variable and a, b, c, d, e, r, α and β are system parameters.

2) ATTRACTOR PHASE DIAGRAM

Set $a = 0.001$, $b = 0.005$, $c = 0.5$, $d = 2$, $e = 4$, $r = 0.01$, $\alpha = 1$, $\beta = 1$, let initial values (0, −4, −0.01, 43), the system attractor of the 4D chaotic system as depicted in Fig.1. Furthermore, Lyapunov exponents are (0.01050, 0, −0.0052, −0.5581), and the Lyapunov dimension is 3.1789. This proves that this system has chaotic characteristics under the above conditions.

3) LYAPUNOV EXPONENT SPECTRUM AND BIFURCATION DIAGRAM

The LEs (Lyapunov exponential spectrum) and BDs (bifurcation diagram) can verify the chaotic behavior, so LEs and BDs under the parameters $d \in [0, 6]$, $r \in [0\ 0.2]$ and $\alpha \in [0\ 1.5]$ are shown in Fig.2(a)-(f), here the iteration step is 0.001. The results show that the new 4D chaotic system has surprising strong randomness.

B. THE RANDOMNESS OF 4D CHAOTIC SYSTEM

In this section, to distinguish the randomness of chaotic systems, NIST SP 800-22 is introduced [54]. The results of the test can be judged by three criteria. The first evaluation criterion is $P - value$, set the confidence level is α, if $P - value > 0.01$, it can be proved that the sequence of this system can pass the randomness test. The second evaluation criterion is $P - value_T$, which can be used to prove that all $P - value$ sample blocks obey uniform distribution. The

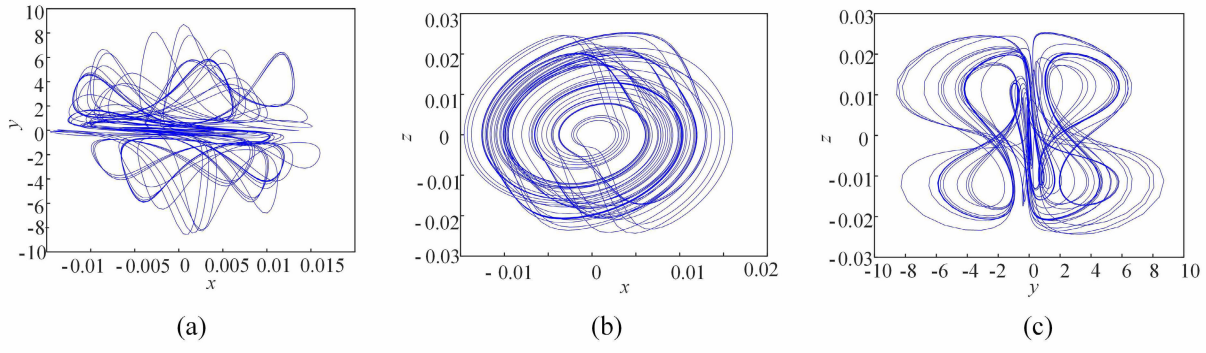

FIGURE 1. Phase diagram display on three planes (a) x-y plane (b) x-z plane (c) y-z plane.

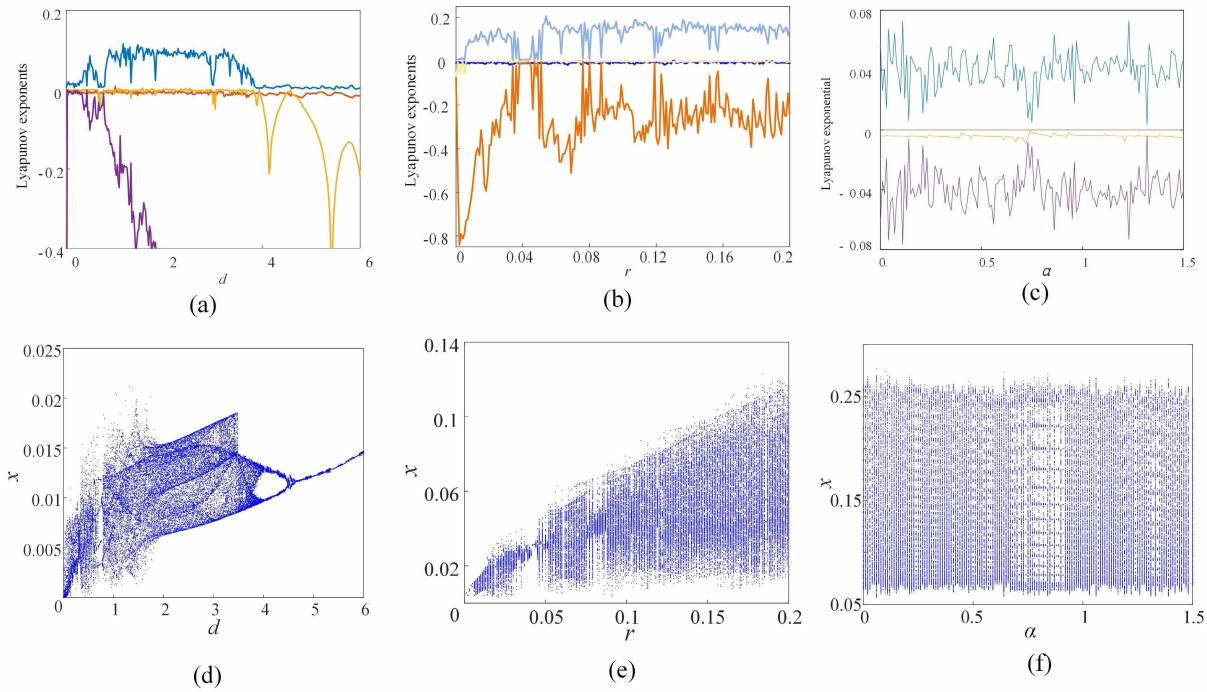

FIGURE 2. LEs and BDs of the 4D chaotic system (a) LEs with $d \in [0, 6]$ (b) LEs with $r \in [0, 0.2]$ (c) LEs with $\alpha \in [0, 1.5]$ (d) LBs with $d \in [0, 6]$ (e) LBs with $r \in [0, 0.2]$ (f) LBs with $\alpha \in [0, 1.5]$.

formula is as follows

$$\chi^2 = \sum_{i=1}^{10} \frac{(G_i - 0.1s)^2}{0.1m} \tag{2}$$

where G_i represents the number of $P - value$ in sub-interval i, and m represents the sample size. If the proved chaotic sequence has a high degree of randomness, then $P - value_T \geq 0.0001$.

The third criterion is the rate at which the sample sequence of the test passes. First set the confidence level α, and then the confidence used for the test results is defined as follows

$$\hat{P} = \sqrt{\frac{\hat{P} \cdot (1 - \hat{P})}{s}} \tag{3}$$

where $\hat{P} = 1 - \alpha$, s represents the size of the test sample. For $P - value \geq 0.01$, the ideal pass ratio should be 0.9960. Tab.1 shows the randomness test of the 4D chaotic system. The results show that the system can pass the test and the generated sequence has good randomness.

C. DNA INFORMATION

1) DNA ENCODING AND DECODING RULES

The concept of DNA (deoxyribonucleic acid) originated from biology and represents the genetic information of biological characteristics. It contains 4 types of nitrogenous bases, namely adenine (A), guanine (G), cytosine (C) and thymine (T). A double-stranded DNA molecule has a regular double helix structure in which hydrogen bonds link two complementary bases. The connected complementary base pairs are respectively (A, T) and (C, G). In binary coding, in order to link binary and DNA coding, assuming that binary 00 and 01, 10 and 11 are complementary, then 8 DNA encoding

TABLE 1. The randomness test results of the 4D chaotic system.

Test Name		Test sequence x	
		$P-value_T$	Pass Rate
Frequency		0.350485	0.99
Block Frequency		0.137282	0.99
Runs		0.514124	0.97
Longest Run		0.350485	1
FFT		0.616305	0.98
Universal		0.419021	1
Approximate Entropy		0.574903	0.99
Linear Complexity		0.419021	1
Non Overlapping Template		0.000406	0.99
Overlapping Template		0.171867	1
Cumulative Sums	Forward	0.699313	0.98
	Reverse	0.514124	0.99
Serial	$P-value_1$	0.991468	0.98
	$P-value_2$	0.616305	1
Random Excursions		0.509254	0.99
Random Excursions Variant		0.445963	0.99

TABLE 2. Addition and subtraction rules.

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

and decoding rules as listed in Tab.2. The pixels of the gray image may be converted to 8 bits. For example, the pixel value is 149, which is represented as an 8-bit binary code 10010101. If rule 5 in Tab.2 is used for encoding, the binary sequence can be converted into encoding ATTT, and the binary sequence 10010101 can be recovered when rule 5 is selected for decoding.

2) DNA OPERATIONS

Binary arithmetic rules include addition, subtraction and XOR operations. The corresponding encoded DNA sequences also have the same operations. The rules of DNA operations are given in Tab.3 and Tab.4, respectively.

3) DNA HAMMING DISTANCE

DNA hamming distance is introduced based on the principle of hamming distance. The calculation formula is

$$\begin{cases} H(x,y) = \sum_{i=1}^{n} h(x_i, y_i) \\ \sum_{i=1}^{n} h(x_i, y_i) \begin{cases} 0, & x_i = y_i \\ 1, & x_i \neq y_i \end{cases} \end{cases} \tag{4}$$

where x_i and y_i are elements of two DNA sequences x and y with the same size, if $x_i = y_i$, $h(x_i, y_i) = 0$; otherwise, $h(x_i, y_i) = 1$, $H(x, y)$ is the sum of $h(x_i, y_i) = 1$.

D. BINARY TREE

1) IMPROVED BINARY SORT TREE

Binary Sort Tree (BST) is also called Binary Search Tree. Here, we limit the number of nodes in the tree to three, assuming that the three nodes are represented as A, B, and C.

TABLE 3. DNA XOR rules.

XOR	A	G	C	T
A	A	G	C	T
G	G	A	T	C
C	C	T	A	G
T	T	C	G	A

TABLE 4. Addition and subtraction rules.

+	A	C	G	T	-	A	C	G	T
A	A	C	G	T	A	A	T	G	C
C	C	G	T	A	C	C	A	T	G
G	G	T	A	C	G	G	C	A	T
T	T	A	C	G	T	T	G	C	A

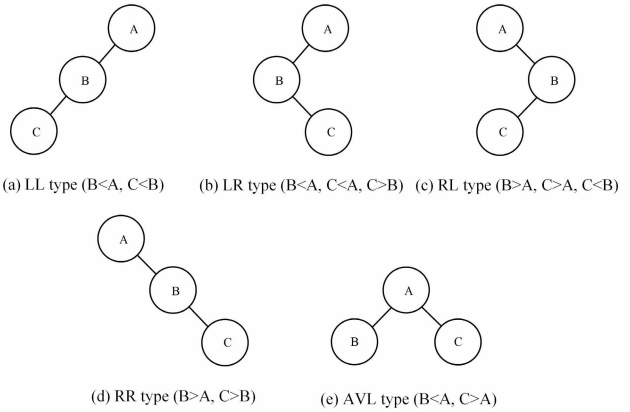

FIGURE 3. Five types of IBST.

Therefore, Improved Binary Sort Tree (IBST) has five basic types, as shown in Fig.3.

2) IMPROVED BALANCED BINARY TREE

Balanced binary tree (AVL), the absolute value of the height difference between the left and right subtrees of the tree does not exceed 1. For the above-mentioned IBST with only three nodes, there are five rules for adjusting to improved balanced binary tree (IAVL). The specific implementation steps are as follows

1. LL type balanced rotation (right rotation): that is, perform a right rotation operation on the IBST in the shape of Fig.3(a), rotate B upward instead of A as the root node, and rotate node A to the lower right to become the right child node of B. The operation process is shown in Fig.4(a).

2. LR type balanced rotation (first left and then right double rotation): That is, two rotation operations are required for the IBST of the shape of Fig.3(b), first left rotation and then right rotation, and node C is rotated left as the left child of B. Exchange the positions of nodes B and C. Rotate node A to the lower right to become the right child node of B. The operation process is shown in Fig.4(b).

3. RL type balanced rotation (right and then left double rotation): that is, two rotation operations are required for the IBST of the shape of Fig.3(c), first right rotation and then left

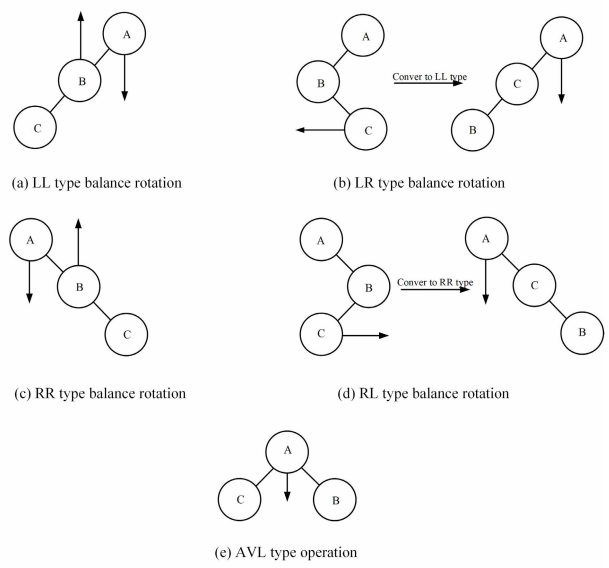

FIGURE 4. IAVL adjustment rules.

rotation, and node C is rotated right as the right child of B. Exchange the positions of nodes B and C. Rotate the node A to the lower left to become the left child node of C. The operation process is shown in Fig.4(c).

4. RR type balanced rotation (left rotation): that is, a left rotation operation is required for the IBST of the shape of Fig.3(d), B is rotated to the upper left instead of A as the root node, and node A is rotated to the lower left to become the left child node of B. The operation process is shown in Fig.4(d).

5. AVL type: In order to facilitate the operation of scrambling the image, A nodes in Fig.3(e) is adjusted downward, and exchange the positions of nodes B and C in Fig.3(e). The result is shown in Fig.4(e).

III. ENCRYPTION SCHEME

A. GENERATION OF THE INITIAL VALUES OF THE CHAOTIC SYSTEM

Firstly, SHA 256 hash function of the plain image is used to generate to 256-bit external secret key. Then, divide the external secret key K into 32 groups, each group contains 8 bits as follows

$$\begin{cases} K = k_1, k_2, \ldots, k_{32} \\ k_i = \{k_{i,0}, k_{i,1} \cdots, k_{i,7}\} \end{cases} \tag{5}$$

where, $i = 1, 2, 3, \ldots, 32$, k_i means the ith bit group.

The parameters used to generate initial values can be obtained as follows

$$\begin{cases} m_a = \mathrm{mod}(10^{-15} mean(K(1:8) \oplus K(9:16), 2^6) \\ m_b = \mathrm{mod}(10^{-15} mean(K(17:24) \oplus K(25:32), 2^6) \\ m_c = \mathrm{mod}(10^{-15} sum(K(1:8) \oplus K(9:16) \oplus \\ \qquad K(17:24), 2^6) \\ m_d = \mathrm{mod}(10^{-15} sum(K(1:8) \oplus K(9:16) \oplus \\ \qquad K(17:24 \oplus K(25:32), 2^6) \end{cases} \tag{6}$$

where mod is the modular operator, mean () is used to average, and $K_{2i} \oplus K_{2i+1}$ is the XOR operation between K_{2i} and K_{2i+1}.

The initial values (x_0, y_0, z_0, u_0) of the 4D chaotic system are given as keys, and perform addition operation with (k_a, k_b, k_c, k_d) to obtain new initial values of the chaotic system as follows

$$\begin{cases} x_1 = x_0 + m_a \\ y_1 = y_0 + m_b \\ z_1 = z_0 + m_c \\ u_1 = u_0 + m_d \end{cases} \tag{7}$$

B. PERMUTATION OPERATION BASED ON IAVL

A new scrambling algorithm can be designed based on the adjustment process of the IAVL. The idea is as follows, assuming that the elements used to construct the IBST come from the matrix Q of the same size as the image, in the process of adjusting IBST to IAVL involves the movement and exchange of nodes, corresponding a new scrambling algorithm can be associated with it. The specific scrambling operations are as follows

Step 1: Suppose the image P size is $M \times N$, give the initial value (x_1, y_1, z_1, u_1) of the chaotic system, which produced by formula 7, and iterate the chaotic system $M \times N + l$ times, discard the first l times to obtain state variables x_i, y_i, z_i and u_i, then the chaotic sequence S z $x_1, x_2, \ldots, x_{MN}$.

Step 2: Modify the chaotic sequence S, and turn it into a matrix of size $M \times N$.

$$S = \mathrm{mod}(floor((x_i + 100) \times 10^{10}),\ 10 \times \max(M, N)) + 1 \tag{8}$$

Sort S to generate the index value Q as shown in formula 9.

$$[lx, Q] = sort(S) \tag{9}$$

Step 4: Take the ith row element $Q(i,:)$ ($i = 1, 2, 3, \ldots, M$) of the matrix Q, and make a group of every three elements (a total of $N/3$ groups), and write down where the three elements are column j, $j+1$ and $j+2$ ($j = 1, 4, 7, \ldots, N/3\text{-}2$). Construct a IBST respectively, where A = $Q(i, j)$, B = $Q(i, j+1)$ and C = $Q(i, j+2)$. Then, according to the adjustment process of IAVL, perform column cyclic shift and column exchange operations on the image. The specific rules are as follows

Case1: If the type to be adjusted is the LL type, according to Fig.4(a), the column j where the element of the A node is located corresponds to the jth column of the image and moves downward $Q(i, j)$ times, and the column $j+1$ where the element of the B node is located corresponds to the jth column of the image and moves downward $Q(i, j+1)$ times.

Case2: If the type to be adjusted is the LR type, according to Fig. 4(b), move the ith row of the image to the left by $Q(i, j+2)$ times, and the value of the C node element $Q(i, j+2)$ and B node element $Q(i, j+1)$ are regarded as two columns in the image to exchange, and then the j column where the A node

element is located corresponds to the jth column of the image and moves down $Q(i, j)$ times.

Case3: If the type to be adjusted is the RL type, according to Fig.4(c), move the ith row of the image to the right by $Q(i,j+2)$ times, and the value of the C node element $Q(i,j+2)$ and B node element $Q(i, j+1)$ are regarded as two columns in the image to exchange, and then the j column where the A node element is located corresponds to the jth column of the image and moves down $Q(i, j)$ times.

Case4: If the type to be adjusted is the RR type, according to Fig.4(d), the column j where the element of the A node is located corresponds to the jth column of the image and moves downward $Q(i, j)$ times, and the column $j+1$ where the element of the B node is located corresponds to the jth column of the image and moves downward $Q(i, j+1)$ times.

Case5: If the IBST is constructed as shown in Fig.4(e), the column j where the element of the A node is located corresponds to the jth column of the image and moves downward $Q(i, j)$ times, and the value of the C node element $Q(i, j+2)$ and B node element $Q(i, j+1)$ are regarded as two columns in the image to exchange, and then the j column where the A node element is located corresponds to the jth column of the image and moves down $Q(i, j)$ times.

Step 5: Return to Step 4, $i = i+1$ and repeat M-1 times to get the scrambled matrix $B2$. In order to facilitate the understanding of the entire scrambling operation process, we take one step in the scrambling process as an example. Assuming that $Q(1,:)$ is (1, 6, 3, 2, 4, 5) and image P is (6×6) matrix, as shown in Fig.5(a), the IBST constructed by taking the first three numbers of $Q(1,:)$ is shown in Fig.3(c) (A = 1, B = 6, C = 3), and adjust the corresponding operation rules to IAVL as Case4: Here the adjustment type is RL operation, as depicted in Fig.4(d) and $i = 1, j = 1$. First, since C = 3 and the line to be operated is $i = 1$, the first line of the image is shifted to the right circle once, as shown in Fig.5(b); Second, swap columns 3 and 6 of the image, since the nodes B = 6 and C = 3 have to be swapped; Finally, since A = 1 and $j = 1$, the first column of the image is circularly shifted downward once; The final result is illustrated in Fig.5(c).

C. BLOCK-BASED DYNAMIC DNA ENCODING

In the encoding process, block-based DNA coding is proposed with the help key matrix to achieve dynamic coding. That is to say, the coding rules of the image block are selected by the corresponding key matrix block. Here, we update the initials value of the chaotic system again by using the DNA hamming distance algorithm on the plain image. When the plain image changes, the corresponding key matrix will also change, and finally change the choice of image coding rules. The design of the coding rules is demonstrated as follows.

Step 1: Each pixel of the image can be encoded to produce four DNA codes where coding rule $u = 1$ is selected. Then, taking the first DNA code of all image pixels can form a DNA matrix $BP0$ of size $M \times N$, then manipulating the same operation to the second, third and fourth DNA codes of the image to generate $BP1$, $BP3$ and $BP4$.

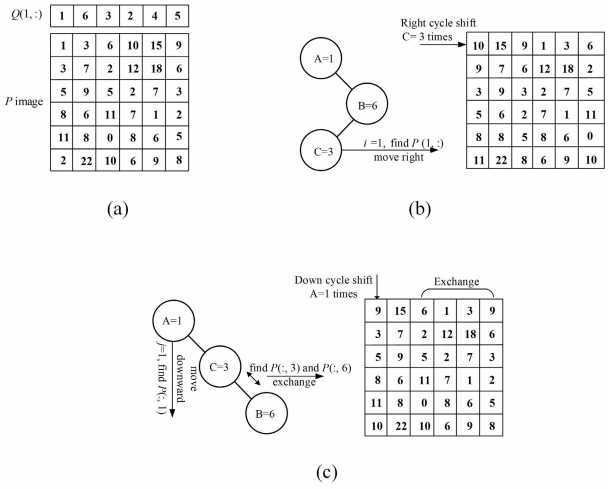

FIGURE 5. Image scrambling example.

Step 2: The DNA hamming distances are computed using the four DNA matrices, the result is as follows

$$\begin{cases} d_1 = H(BP0, BP1) \\ d_2 = H(BP0, BP2) \\ d_3 = H(BP0, BP3) \\ d_4 = H(BP1, BP2) \end{cases} \tag{10}$$

Step 3: Set $r = M \times N$, initial values after update of the 4D chaotic system are defined as follows

$$\begin{cases} x_1 = x_0 + 0.5 \times d_1/r \\ y_1 = y_0 + 0.6 \times d_2/r \\ z_1 = z_0 + 0.6 \times d_3/r \\ u_1 = u_0 + 0.5 \times d_4/r \end{cases} \tag{11}$$

Step 4: Iterated Eq. (1) $r+ l$ times using the new initial values in (9), then discard the former l values, modify the obtained state x_i, y_i and z_i variables as follows

$$\begin{cases} x'_i = \dfrac{sum(abs(x_i) + floor(y_i))}{r} \\ y'_i = \dfrac{sum(abs(y_i) + floor(z_i))}{r} \\ z'_i = \dfrac{sum(abs(x_i) + floor(u_i))}{r} \end{cases} \tag{12}$$

Step 5: The final key matrix can be obtained by the following formulas

$$K2 = \mathrm{mod}(round(\frac{3 \times 10^8 \times T}{F}), 256) \tag{13}$$

where

$$\begin{cases} T(i,j) = (x'_i + i)^2 + (y'_i + i)^2 + {z'_i}^2 \\ F(i,j) = (x'_i + i)^2 + j^3 + y'_i \times i \times j \times j \end{cases} \tag{14}$$

where T (i,j) and $F(i,j)$ represent the values of a new matrix in row i, column j, which is used to calculate the key matrix $K2$.

TABLE 5. DNA operation rules table.

Rule	A	T	C	G
f (,)	DNA XOR	DNA addition	DNA subtraction	DNA XOR

Step 6: Divide image $B2$ and the key $K2$ into $M\times N/4$ blocks of size 2×2, each block encoding rule s_i comes from the following formulas

$$s_i = \mathrm{mod}(floor((m_1 + m_2 + m_3 + m_4 + m_5 + m_6)/6), 8) + 1 \tag{15}$$

where

$$\begin{cases} m_1 = abs(K2_i(1) - K2_i(3)) \\ m_2 = abs(K2_i(1) - K2_i(2)) \\ m_3 = abs(K2_i(1) - K2_i(4)) \\ m_4 = abs(K2_i(3) - K2_i(2)) \\ m_5 = abs(K2_i(3) - K2_i(4)) \\ m_6 = abs(K2_i(2) - K2_i(4)) \end{cases} \tag{16}$$

where $K_{2i}(k)$ $(k = 1, 2, 3, 4)$ represents the kth element of the ith $(i = 1, 2, \ldots, M\times N/4)$ block of the key matrix, m_j $(j = 1, 2, \ldots, 6)$ represents the absolute value of the difference between any two elements in the block.

Step 7: DNA matrix D with a size of $M\times N\times 4$ is obtained according to s_i coding $B2$ and manipulate the same encoding operation to $K2$, then $K3$ is obtained.

D. INTRA-BLOCK AND INTER-BLOCK DIFFUSION OPERATION

In the diffusion process, the diffusion operation is carried out with the help of the key matrix produced by Sec. III.C. Diffusion steps include intra-block and inter-block diffusion, firstly, intra-block DNA sequence operations include 3 types in Tab.4, it depends on the corresponding DNA sequence in the key block. Finally, the algorithm between blocks is a DNA addition operation and the diffusion operation is carried out in the horizontal and vertical directions respectively. The specific steps are shown below.

Step 1: Divide DNA matrix D and $K3$ into $M\times N/4$ blocks of size 2×2.

Step 2: Selected rules functions f (,) are defined by the following Tab.5. If the first element of each block is equal to 'A', 'T', 'C' or 'G', the corresponding DNA operation is selected.

Step 3: Intra-block diffusion is implemented by the following formula

$$C_i(k) = f(f(D_i(k), K3_i(k)), C_i(k-1)) \tag{17}$$

$C_i(k)$ $(k = 1, 2, 3, 4)$ represents the kth element of the ith $(i = 1, 2, \ldots, M\times N/4)$ block of the encrypted, D_i (k) and $K_{3i}(k)$ are the kth element of the corresponding encoded DNA image block and key DNA block, respectively. and $C_i(k\text{-}1)$ represents the $(k\text{-}1)$th element of the encrypted in the block. When $k = 1$, $C_i(0) = D_i(4)$.

Step 4: Divide C into N blocks of size $M\times 4$, the horizontal diffusion between blocks is as follows

$$C1(i+1) = C(i) + C(i+1) \tag{18}$$

where $C(i)$ represents the ith $(i = 1, 2, \ldots, N)$ block in the horizontal direction, $C(i{+}1)$ represents the block behind $C(i)$ and $C1(i{+}1)$ indicates the result after diffusion. When $i = N$, $C1(1) = C(1) + C(N)$.

Step 5: Divide $C1$ into $M/4$ blocks of size N, the vertical diffusion between blocks is as follows

$$C2(i+1) = C1(i) + C1(i+1) \tag{19}$$

where $C1(i)$ represents the ith $(i = 1, 2, \ldots, M/4)$ block in the vertical direction and $C1(i{+}1)$ represents the block behind $C1$(i), $C2(i{+}1)$ indicates the result after diffusion. When $i = M/4$, $C2(1) = C1(1) + C1(M/4)$ and it means that the diffusion is completed here.

E. THE WHOLE ENCRYPTION PROCESS

As you can see from the Fig.6, the complete encryption step is divided into six steps as shown below.

Step 1: Input standard gray image B with a size of $M\times N$.

Step 2: Calculate the external key K determined by image B through SHA 256 function in Eq. (5). The initial values of the 4D chaotic system can be calculated by using Eq. (7).

Step 3: The sequence S is generated by using new initial values in Eq. (7), and perform permutation operation on the image B of each pixel as stated in Sec. II.B to get permutated matrix $B2$.

Step 4: The key matrix $K2$ is generated by using new initial values in Eq. (11) whose lengths are M and N, respectively. Partition image $B2$ into $M\times N/4$ blocks of size 4×4 and encode them using dynamic coding technology as shown in Sec. III.C to obtain DNA matrix D and $K3$ of size $M\times N\times 4$.

Step 5: Implement intra-block and inter-block diffusion on the encoded image D to get diffusion $C2$ with the help of the key matrix $K3$ as described in Sec. III.D

Step 6: Decode the DNA matrix $C2$ by the f decoding rule and it can be obtained through f = (floor(mod($d_1\times 1000$),8+1)) where d_1 came from Eq. (10). Finally, the cipher image is obtained.

IV. SIMULATION RESULTS

In this section, the simulation tests are implemented on a computing device with the following hardware and software environments: intel core i5-105G3 processor and 16GB (DDR4 3200MHz) RAM, MATLAB 2019a and window-10 system. In our algorithm, the initial values and parameters of the 4D chaotic system are chosen as: ($x_0 = 0$, $y_0 = -4$, $z_0 = -0.01$, $u_0 = 43$) and ($a = 0.001$, $b = 0.005$, $c = 0.5$, $d = 2$, $e = 4$, $r = 0.01$, $\alpha = 1$, $\beta = 1$), respectively and iteration parameter $l = 500$. Finally, select a series of (256×256) standard grayscale images, such as House, Pepper, Barbara and Baboon as the encryption objects. The simulation images are depicted in Fig.7.

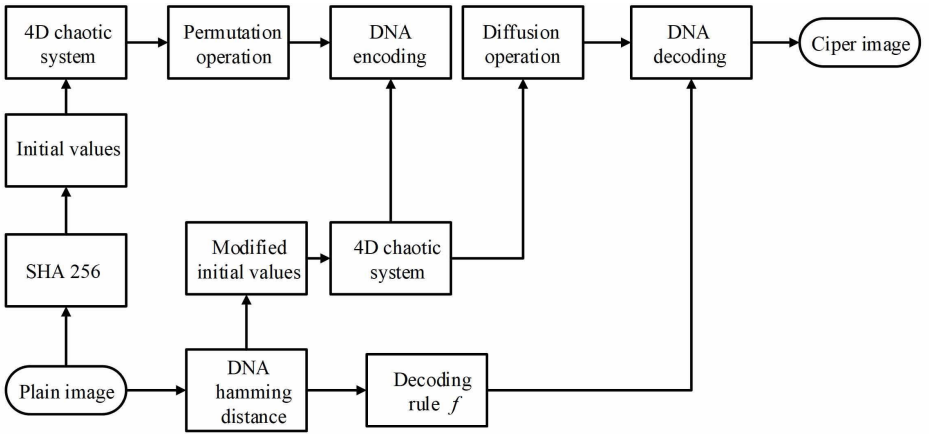

FIGURE 6. Complete encryption flow chart.

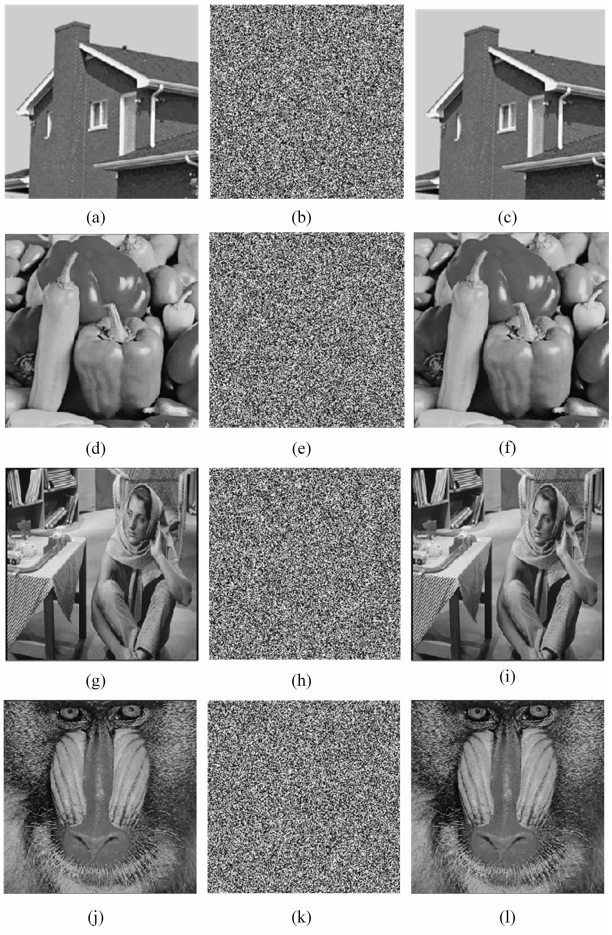

FIGURE 7. The experiment test results (a) (d) (g) (j) are plain image House, Pepper, Barbara and Baboon (b) (e) (h) (k) are cipher image House, Pepper, Barbara and Baboon (c) (f) (i) (l) are decrypted image of House, Pepper, Barbara and Baboon.

V. SECURITY ANALYSES

A. KEY SPACE ANALYSIS

The primary goal of the encryption system is to effectively deal with brute force attacks. After research by cryptography experts, the length of the key is at least 100 bit [4]. In the proposed algorithm, the composition of the key can be expressed as the following

(1) The initial values and control parameters are (x, y, z, u) and $(a, b, c, d, e, r, \alpha, \beta)$.

(2) The 256-bit hash values are given by SHA 256 function to generate new initial values for the permutation phase.

(3) The four disturbance $(d_1, d_2, d_3, d_4,)$ parameters are obtained using DNA hamming distance to get new initial values for DNA encoding and diffusion phase.

(4) In order to avoid transit effects, iterating parameter l (500) is given as secret key.

(5) DNA encoding u and decoding rule f.

When the computer's precision is set to 10^{-15}, the key size of the initial value and parameters of the chaotic system will be $10^{180} \approx 2^{598}$, the external key space is 2^{128}, key space for other parameters will reach $10^{90}+2^9+2^6 \approx 2^{207}$. So, the total key space is about 2^{933}, Tab.6 shows the comparison of key spaces in different literatures. It can be seen that our algorithm has the largest key space, so the system will not be damaged by violent attacks. At the same time, in the face of the management of such a large key space, we use the parameters mentioned in (1) and (4) as the encryption and decryption side private keys. (2), (3) and (5) are transmitted on the key channel. This can greatly reduce the administrative burden of the key space.

TABLE 6. Comparison of key spaces.

Encryption scheme	Our algorithm	Ref.[24]	Ref.[33]	Ref.[34]
Key space	2^{315}	2^{526}	2^{580}	2^{598}

B. HISTOGRAM ANALYSIS

The histogram can present the statistical information of the pixels graphically. A good encryption system should resist statistical attacks and it has a uniform distribution. Fig.8(a) and (b) are the histograms of the image Pepper (256×256) before and after encryption. After observing Fig.8(b), since

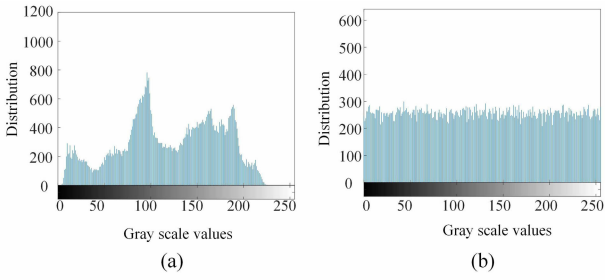

FIGURE 8. Histogram of the plain Pepper and it cipher image (a) histogram of the plain image Pepper (b) histogram of cipher image.

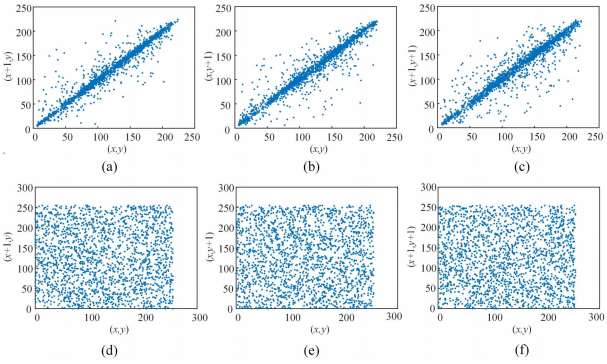

FIGURE 9. Distribution of correlation (a) (b) (c) correlation of Pepper image in H, V and D directions (d) (e) (f) correlation of the cipher image in H, V and D directions.

the image is quite flat, the attacker cannot obtain any information just by observing the image.

C. CHI-SQUARE AND VARIANCE TESTS

In order to further evaluate the distribution uniformity of pixels, the Chi-square test will be used as a test tool [2]. It is expressed as

$$\chi^2_{text} = \sum_{n=0}^{255} \frac{(F_i - G_i)^2}{G_i} \tag{20}$$

where F_i and G_i represent observed frequency distribution and theoretical frequency distribution, respectively and $G = MN/256$.

The smaller the chi-squared value, the more uniform the histogram. In other words, the better the encrypted image information is hidden. In general, the commonly used significance level is 0.05, $\chi^2_{0.05} = 293.2783$. Through the Chi-square test on Fig.8, the test result is 256.0156, which is far lower than $\chi^2_{0.05}$, so Fig.8 can be considered to be approximately evenly distributed.

The variance is another kind of quantitative analysis of histogram [32]. The smaller the variance, the more uniform the histogram is tested. Its mathematical expression formula is

$$Var(C) = \frac{1}{N^2} \sum_{i=0}^{N} \sum_{j=0}^{N} \frac{1}{2} \times (c_i - c_j)^2 \tag{21}$$

where N is the number of gray levels of an image. For an 8-bit gray image, $N = 256$. C is a vector and $C = c_0, c_1, \ldots, c_{N-1}$, c_i and c_j are the numbers of pixels with gray values equal to i and j, respectively. The variance test is performed on the histogram in Fig.8 and the result is $Var(C) = 256.0156$. In the variance test, the difference around 260 is already ideal, that is, the histogram near this value is very uniform.

D. CORRELATION ANALYSIS

The adjacent pixels generally have a certain connection, which is often used by attackers to analyze image information. The 2000 pairs of adjacent points in the image are selected to test the correlation, where the selection directions of the pixel pairs are horizontal (H), vertical (V) and diagonal (D) directions. The correlation coefficient (r_{xy}) calculation formulas are as follows

$$r_{xy} = \frac{cov(x, y)}{\sqrt{D(x)D(y)}} \tag{22}$$

$$cov(x, y) = E\{[x - E(x)][y - E(y)]\} \tag{23}$$

$$D(x) = \frac{1}{N} \sum_{i=1}^{N} [x_i - E(x)]^2 \tag{24}$$

$$E(x) = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{25}$$

where x and y is a pair of adjacent pixel values, N represents the number of image pixels, $E(x)$ is expectation, $\text{cov}(x, y)$ and $D(x)$ represent the covariance and variance, respectively.

Fig.9 shows a quantitative representation of the correlation before and after image encryption and the specific values of the correlation and the comparison with other documents as listed in Tab.7. It can be seen from Fig.9 and Tab.7 that the correlation in the original image is an approach to 1, but the encrypted image is an approach to 0. Compared with [18], [42], [43], our algorithm makes the correlation coefficient of the encrypted image lower, while compared with [49], the correlation coefficient is close, which shows that the correlation of the plain image is eliminated after encryption.

E. INFORMATION ENTROPY

It is generally believed that information entropy is a measure of the degree of randomness of information. The calculation formula entropy is as follows

$$H(m) = \sum_{i=0}^{L-1} p(m_i) \log \frac{1}{p(m_i)} \tag{26}$$

where m_i represents the source value, $p(m_i)$ is the probability of the source value. L is equal to 256 due to the range of the pixel value belongs [0 255] for grayscale images.

For the encrypted image, the information entropy should be close to the ideal value 8. Tab.8 and Tab.9 list the information entropy of an image before and after encryption and

TABLE 7. Correlation comparison of different literature.

Image	Direction	Plain image	Our algorithm	Ref.[18]	Ref.[42]	Ref.[43]	Ref.[49]
House	Horizontal	0.9238	-0.0018	0.0085	0.0096	0.0053	0.0027
	Vertical	0.9701	0.0051	0.0093	0.0216	0.0151	0.0045
	Diagonal	0.9082	0.0080	0.0198	0.0031	0.0228	0.0110
Baboon	Horizontal	0.8658	-0.0078	0.0208	0.0023	0.0082	0.0005
	Vertical	0.8121	-0.0032	0.0078	0.0360	0.0171	0.0069
	Diagonal	0.7604	-0.0013	0.0184	0.0191	0.0257	0.0024
Pepper	Horizontal	0.9425	-0.0017	0.0050	0.0068	0.0199	0.0030
	Vertical	0.9466	0.0040	0.0016	0.0178	0.0087	0.0066
	Diagonal	0.9143	-0.0035	0.0144	0.0093	0.0137	0.0060
Barbara	Horizontal	0.9440	0.0068	0.0054	0.0136	0.0036	0.0047
	Vertical	0.9651	-0.0003	0.0222	0.0138	0.0114	0.0084
	Diagonal	0.9229	0.0093	0.0172	0.0051	0.0130	0.0001

TABLE 8. Comparison of information entropy results with [18], [42], [43], [49].

Image	Size	Plain image	Our algorithm	Ref.[18]	Ref.[42]	Ref.[43]	Ref.[49]
House	(256×256)	7.4444	7.9973	7.9974	7.9880	7.9954	7.9977
Baboon	(256×256)	7.0091	7.9972	7.9972	7.8778	7.9805	7.9970
Pepper	(256×256)	7.5797	7.9972	7.9968	7.9831	7.9951	7.9971
Barbara	(256×256)	7.5838	7.9973	7.9970	7.9056	7.9942	7.9968

TABLE 9. Comparison of information entropy results with [35]–[37].

Image	Size	Plain image	Our algorithm	Ref.[35]	Ref.[36]	Ref.[37]
House	(256×256)	7.0097	7.9973	7.9972	7.9974	-
Baboon	(512×512)	7.2925	7.9972	-	7.9973	7.9993
Pepper	(512×512)	7.7624	7.9993	7.9993	-	7.9993

TABLE 10. Local entropy test results.

Image	Size	Our algorithm	Significance level		
			0.001	0.01	0.05
House	(256×256)	7.9026	Pass	Pass	Pass
Barbara	(256×256)	7.9032	Pass	Pass	-
Baboon	(512×512)	7.9024	Pass	Pass	Pass
Pepper	(512×512)	7.9031	Pass	Pass	-

compares it with related references. In Tab.8, it can be seen that the information entropy of our algorithm is only lower than that of House in [42] and [49]. Then by comparing the information entropy of images of different sizes in Tab.9, it can be seen that the information entropy of images of (256 × 256) and (512 × 512) is always around 7.9973 and 7.9993, which is basically the same as that of [35]–[37]. Therefore, the encrypted image almost does not have the possibility of information leakage.

The Local Shannon entropy can be used to measure the precise randomness of the image pixels after encryption, so the security of our algorithm information entropy can be further demonstrated. The formula is as follows

$$\overline{H_{k,T_B}(S)} = \sum_{i=1}^{k} \frac{H(s_i)}{k} \tag{27}$$

where k is the number of non-overlapping blocks that the source S is divided into, T_B is the number of pixels contained in each block, and $H(S_i)$ is the information entropy of each block. In the test of local entropy in this paper, parameters $k = 30$ and $T_B = 1936$ are set. At different significance level (0.001, 0.01, 0.05), the local Shannon entropy should be between (7.901515698, 7.903422936), (7.901722822,7.903215812) and (7.901901305, 7.903037329), respectively [18]. As can be seen from Tab.10, all images pass the security test at the confidence level of 0.001 and 0.01, while only a part fails at the confidence level of 0.05. Therefore, our algorithm can pass the security test of information entropy.

F. DIFFERENTIAL ATTACK

To put it simply, differential attack tests the sensitivity of encrypted images to changes in plain images. At present, NPCR (pixel change rate) and UACI (uniform mean change intensity) are the most commonly used differential attack test tools. They can be expressed as follows

$$\text{NPCR} = \frac{\sum_{i,j} D(i,j)}{L} \times 100\% \tag{28}$$

$$\text{NPCR} = \frac{\sum_{i,j} D(i,j)}{L} \times 100\% \tag{29}$$

where C is assumed to be the encrypted image of P, then C_1 is the encrypted image after P randomly changes one pixel, L represents the number of pixels of image P. If $C(i,j) = C_1(i,j)$, $D(i,j) = 0$; Otherwise, $D(i,j) = 1$. We make a slight change to the pixel value at any point of the image in which sizes tested here are all (256×256) and calculate the average value of 10 times of NPCR and UACI as the final comparison

TABLE 11. NPCR and UACI results with [18], [42], [43], [49].

Image		Our algorithm	Ref.[18]	Ref.[42]	Ref.[43]	Ref.[49]
House	NPCR	99.63%	99.57%	99.57%	99.65%	99.60%
	UACI	33.46%	33.51%	33.08%	37.84%	33.50%
Baboon	NPCR	99.62%	99.62%	99.53%	99.62%	99.58%
	UACI	33.46%	33.30%	31.27%	34.91%	33.53%
Pepper	NPCR	99.61%	99.60%	99.61%	99.67%	99.63%
	UACI	33.46%	33.56%	33.42%	35.08%	33.36%
Barbara	NPCR	99.60%	99.59%	99.61%	99.32%	99.60%
	UACI	33.53%	33.42%	31.31%	30.39%	33.57%

TABLE 12. Error metric analysis.

Image	House	Pepper	Baboon	Barbara
PSNR	8.9021	8.617	9.0264	8.7761
MSE	8372.8	8939.4	8136.5	8619.3

result and the results and comparative documents are shown in Tab.11. It is not difficult to find that the NPCR and UACI produced by our algorithm are closer to the expected values of 99.6094% and 33.4653%. Therefore, our scheme has good performance against differential attacks.

G. ERROR METRIC ANALYSES

In this section, PSNR (peak signal to noise ratio) and MSE(mean square error) are introduced to measure the difference between the original image and the cipher image [54]. The smaller the measured PSNR, the larger the difference between the two images before and after encryption. Moreover, the larger the measured MSE, the greater the difference between the two images. The PSNR and MSE formulas are defined as

$$PSNR = 10\frac{Q^2 \times M \times N}{\sum_{i=1}^{M}\sum_{j=1}^{N}(\dot{p}(i,j) - p(i,j))} \tag{30}$$

$$MSE = \frac{1}{M \times N \sum_{i=1}^{M}\sum_{j=1}^{N}(\dot{p}(i,j) - p(i,j))} \tag{31}$$

where $\dot{p}(i,j)$ and $p(i,j)$ represent the two images before and after encryption, and M and N represent the rows and columns of the image respectively.

Tab.12 shows the error analysis results. It can be seen that the values of MSE and PSNR are within the range of expected values after ensuring high-quality encryption of the image. Meanwhile, the mean values of MSE and PSNR of the four images in the table are given, which are 8.83.4, 8556.45 respectively. Therefore, the above results can show that the proposed encryption scheme has high efficiency.

H. ENCRYPTION QUALITY ANALYSIS

Maximum deviation (MD) and irregular deviation (ID) are often used as important indicators to evaluate the quality of encryption [54]. In general, the lower the values of MD and ID, the more uniform the histogram distribution after encryption, and the better the encryption quality. The mathematical formulas for MD and ID are shown below

$$MD = \frac{d1 + d2 - 1}{2} + \sum_{i=1}^{2^b-2} di \tag{32}$$

where d_i represents the quantity difference corresponding to the ith coordinate point of the plain and cipher histograms. b stands for bits.

$$ID = \sum_{i=0}^{2^b-1} |h_i - M_h| \tag{33}$$

$$M_h = \frac{\sum_{i=0}^{2^b-1} h_i}{256} \tag{34}$$

where h_i represents the absolute difference between plain and cipher images.

The deviation from uniform histogram (DH) can alse be used to calculate the quality of the encrypted image, that is, whether the histogram distribution of the encrypted image is uniform [55]. The lower the DH obtained, the better the quality of the encryption. The mathematical formula can be written as

$$DH = \frac{\sum_{c_i=0}^{255} |H_{ci} - H_c|}{m \times n} \tag{35}$$

where $m \times n$ represents the size of the image and H_c represents the histogram of the cipher image. H_{ci} can be obtained by the following formula.

$$H_{c_i} = \begin{cases} \frac{m \times n}{256}, & 0 \le ci \le 255 \\ 0, & elsesshere \end{cases} \tag{36}$$

Tab.13 shows the results of the encryption quality test, in which the tested data are MD, ID and DH. The test image compared with the literature in the table is Pepper. Through comparative analysis, we can see that the data of MD is almost the same, the data of ID is slightly better than ours, but the DH of our algorithm is better. Therefore, our algorithm can be said to have better encryption quality.

I. ENERGY OF ANALYSIS

In the calculation of energy, the GLCM (gray level co-occurrence matrix) is introduced into the calculation,

TABLE 13. Encryption quality test results.

Tests	MD	ID	HD
House	74593	39270	0.0477
Pepper	37234	41890	0.0494
Baboon	72665	43768	0.0484
Barbara	32460	41576	0.0487
Ref.[54]	37651	27168	0.5764

TABLE 14. Energy, contrast and homogeneity tests.

Tests	Energy	Contrast	Homogeneity
House	0.0156	10.5312	0.3880
Pepper	0.0156	10.4241	0.3897
Baboon	0.0156	10.5350	0.3901
Barbara	0.0156	10.4478	0.3908
Average	0.0156	10.4845	0.3884
Ref.[43]	0.0153	10.4520	0.4263

where the GLCM is used to measure the occurrence frequency of two specific pixel values with a certain spatial relationship, and then perform image texture analysis [32]. The test result is usually used to evaluate the information in the image by counting the sum of squares of all the elements in the matrix.

$$Energy = \sum p(i,j)^2 \tag{37}$$

Tab.14 shows the energy value of our encryption scheme. It can be seen that the mean value of energy in the table is close to that of the comparative literature, indicating that the pixels of the encrypted images are disordered, that is, the quality of our encryption algorithm is high.

J. ANALYSIS OF CONTRAST

The difference intensity of adjacent pixels can be calculated by contrast analysis. For an encrypted image, the better the encryption quality, the higher the contrast value [32]. The mathematical formula for contrast is shown below

$$Contrast = \sum |i-j|^2 p(i,j) \tag{38}$$

Tab.14 is the contrast value calculated by our scheme. It can be seen that our algorithm has a high contrast value, so our algorithm can provide high security.

K. HOMOGENEITY

The close relationship between the element distribution in GLCM and the diagonal elements in GLCM can be determined by homogeneity testing. In the evaluation of image encryption, the smaller the value of homogeneity, the higher the quality of the encrypted image [32]. Its mathematical formula is expressed as

$$Homogeneity = \sum \frac{p(i,j)}{1+|i-j|} \tag{39}$$

The values of the test results are collected in Tab.14. It can be seen that our algorithm has a small homogeneity value, which indicates that the proposed encryption scheme is excellent in both encryption quality and security.

L. KEY SENSITIVITY

Key sensitivity is an important index of encryption system, and can be tested in two ways: (i) vastly different cipher images should be obtained when encryption systems use slightly different keys to operate the same original image. (ii) even if the decryption keys are almost the same as the encryption key, the correct decryption image cannot be decrypted.

To test the sensitivity of the encryption process, that is, the first case mentioned above. First, make slight changes to the correct key, then encrypt Baboon (256×256), and finally compare the difference between the encrypted image with the correct key and the wrong key. Here, test values (x_0, y_0, z_0, u_0, a, b, c) as the modified object and all key values are given at the beginning in Sec. III. The amount of change is set to $t = 10^{-12}$, take the modification of x_0 as an example: set $x_0 = x_0 + t$ and keep the rest of the parameters unchanged. Perform the same operation on (y_0, z_0, u_0, a, b, c), and finally seven encrypted images can be obtained, as shown in Fig.10(a)-(g). comparing the difference image of Fig.10(h)-(n), also proves that our encryption system is extremely sensitive to the key.

In addition, to test the key sensitivity in the second case. The key changes in the same way as the decryption process, and the decrypted image is shown in Fig.11. The difference between correctly decrypted and incorrectly decrypted images is presented numerically in Tab.15.

Through a complete test of the sensitivity of the key, even if the attacker uses a nearly correct key, nothing useful can be obtained. This shows that our cryptographic system is extremely sensitive and can pass the key sensitivity test.

M. ROBUSTNESS ANALYSIS

Generally, the image and transmission process will inevitably be affected by noise and the loss of some data. The robustness noise tests include Gaussian white noise pollution and salt paper noise pollution, while the loss of data loss is simulated by cutting the image block.

Pepper image (256×256) is a test image and added with salt and pepper noises to perform noise contaminations with different parameters and the experimental results are presented in Fig.12. To further analyze the impact of noise, NPCR and UACI test values of pepper and salt noise and Gaussian noise under different noise parameters are presented in Tab.16 and 17, respectively. It can be seen from Tab.16 when the noise of pepper and salt is 1%, 3% and 5%, the influence is small, and when the noise is 10%, the influence is greater, but the overall outline is still visible. This shows that the algorithm can pass the salt and pepper noise attack test. although the Gaussian white noise test was slightly worse, the outline of the decrypted image was still clearly visible. It can therefore tolerate a certain degree of Gaussian noise attack. To verify this conclusion, Tab.17 shows the test results of Gaussian noise under different parameters. By analogy with the literature [34] and [36], We find that the performance of NPCR and UACI in the table is not as good as [34], but [36] compared to our NPCR is slightly larger but

TABLE 15. Key sensitivity test results.

Figures	Key values							Diffrence values	
	x	y	z	u	a	b	c	NPCR	UACI
8(i)	$0+10^{-12}$	-4	-0.01	43	0.001	0.005	0.5	99.61%	33.48%
8(j)	0	$-4+10^{-12}$	-0.01	43	0.001	0.005	0.5	99.61%	33.49%
8(k)	0	-4	$-0.01+10^{-12}$	43	0.001	0.005	0.5	99.57%	33.50%
8(l)	0	-4	-0.01	$43+10^{-12}$	0.001	0.005	0.5	99.66%	33.53%
8(m)	0	-4	-0.01	43	$0.001+10^{-12}$	0.005	0.5	99.62%	33.47%
8(n)	0	-4	-0.01	43	0.001	$0.005+10^{-12}$	0.5	99.62%	33.51%
8(o)	0	-4	-0.01	43	0.001	0.005	$0.5+10^{-12}$	99.60%	33.41%

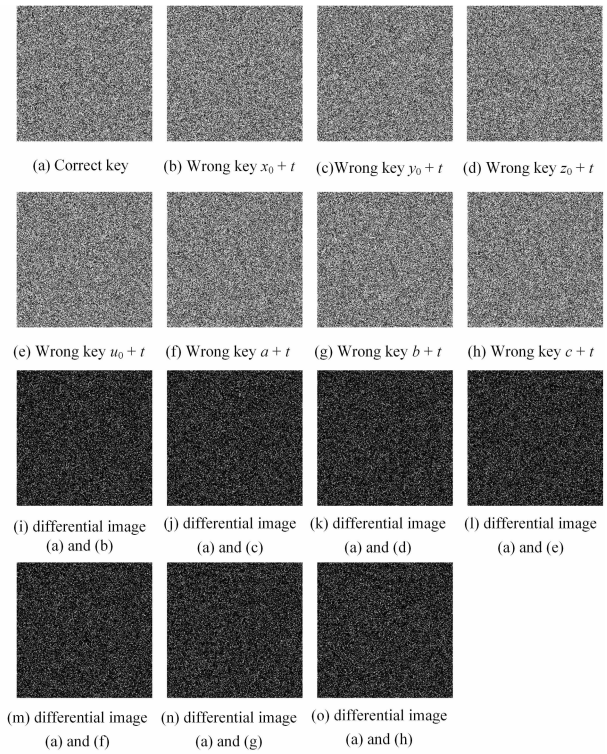

FIGURE 10. Tests of key sensitivity to plain image of Baboon.

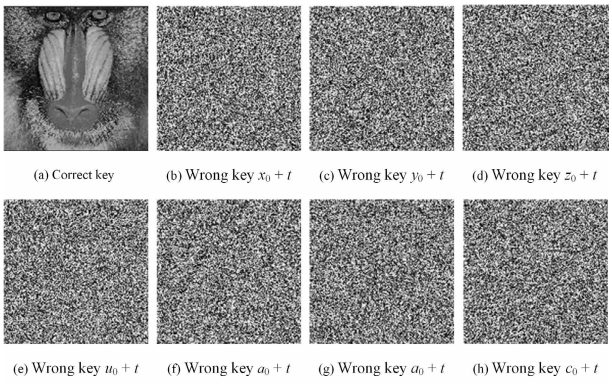

FIGURE 11. Tests of key sensitivity to cipher image of Baboon.

the UACI is smaller. From this point of view, it also shows that our algorithm has a certain ability to resist Gaussian white noise attack.

TABLE 16. Salt and pepper noise test results.

Noise	1%	3%	5%	10%
NPCR	8.33%	23.48%	36.72%	60.50%
UACI	2.56%	7.24%	11.05%	18.10%

TABLE 17. Gaussian white noise test results.

Variance	0.0001	0.0003	0.0005	0.001
NPCR	95.57%	98.41%	98.82%	99.68%
UACI	15.49%	18.50%	20.045%	22.31%

In block loss operation, the cipher image data loss is set to 1/16, 1/8, 1/4 and 1/2 respectively and the experimental results are given in Fig.13. We can see from Fig.13 that the proposed algorithm can cope with information loss well. In summary, our algorithm has good robustness.

N. KNOWN-PLAINTEXT AND CHOSEN-PLAINTEXT ATTACK ANALYSIS

In general, known - plaintext and chosen - plaintext attacks can easily break encryption systems that are not sensitive to plaintext. They mainly select some special images to obtain encryption system keys and then decrypt the images. In this paper, the generation of the chaotic sequence is very sensitive to the change of plain image, which can be stated from the following two aspects: First, during the scrambling process, SHA 256 participates in the update of the initial value of the chaotic system, which means that once the encrypted image changes, the chaotic sequence will also change. Secondly, the DNA hamming distance used in the diffusion process is also determined by the encrypted image. These two characteristics determine that the encryption system is highly sensitive to the encrypted image, so known - plaintext - attack and known - plaintext attack is invalid to the system.

O. COMPUTATIONAL TIME ANALYSIS

The time consumption of the proposed encryption system is mainly in the following stages: chaotic sequence generation, scrambling operation, dynamic DNA coding and diffusion stage.

In the cryptography system mentioned in this paper, only one 4D chaotic system is used, but we use different initial values to put the chaotic system in the scrambling and diffusion stages, and finally produce two different sets of chaotic

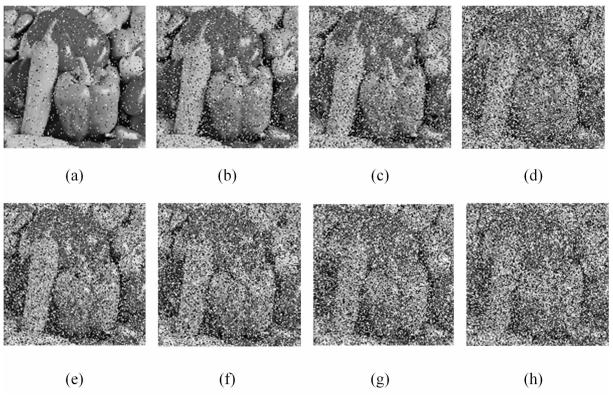

FIGURE 12. Noise tests (a) (b) (c) (d) are salt and paper noise with (0.01, 0.0,3 0.05,0.1) (e) (f) (g) (h) are Gaussian white noise with (0.0001 0.0003 0.0005 0.001).

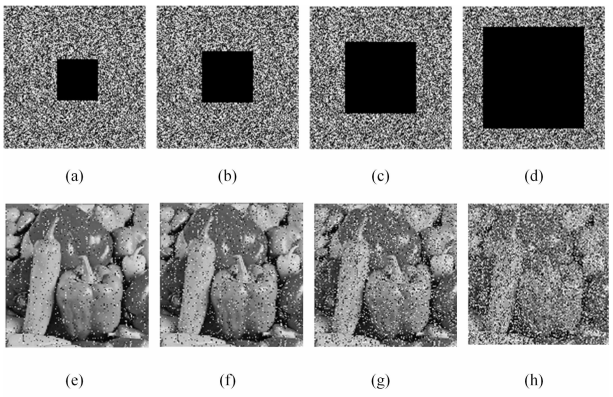

FIGURE 13. Cropping tests (a) (b) (c) (d) are the loss of 1/16 1/8 1/4 1/2 encrypted image blocks, respectively (e) (f) (g) (h) are the decrypted images of (a) (b) (c) (d), respectively.

sequences. The time complexity generated by each group of chaotic sequences is O ($M \times N$), so the total time complexity is O ($M \times N$), where $M \times N$ is the number of pixels in the image.

For the calculation of the time complexity of the scrambling process, the main time consumption is the construction of IAVL, which needs $M \times N/3$ times, and the time complexity is O ($M \times N/3$). Then, the row cycle shift and column swap are carried out for the row and column respectively, so the time complexity is O ($M \times N/3$). For dynamic coding, the cost of time is mainly focused on the selection of coding rules. The encoding rules for each block are determined by the arithmetic operation of a key matrix of the same size (2×2). So the total time complexity is O ($M \times N$).

The diffusion stage can be divided into two stages: intra-block diffusion and inter-block diffusion. The time complexity of intra-block diffusion is O ($M \times N/4$), and the time complexity of inter-block diffusion is O ($M + N/4$). According to the addition rule of time complexity, the time complexity is O ($M \times N/4$). In summary, the time complexity of the algorithm proposed in this paper is O ($M \times N$). Through the analysis of literature [34] and [36], it is found that the time complexity are O ($4 \times M \times N$) and O ($6 \times M \times N$), so our algorithm has certain advantages in time complexity, which may be because the chaotic system in this paper is connected, and the diffusion operation is block diffusion operation.

Finally, by calculating the encryption time of many images (House, Pepper, Barbara, and Baboon) with the size of 256×256 and 512×512, it is found that the time range consumed is 1.114s-1.232s and 5.236s-5.342s, respectively. Therefore, our algorithm has certain advantages in real-time communication.

VI. CONCLUSION

In this paper, a new 4D chaotic system is applied to image encryption for the first time. It has been proved rich dynamical behaviors through the analysis of phase diagram, LEs and BDs, so it is suitable for chaotic image encryption. In the permutation process, by observing the adjustment process of IAVL, a scrambling algorithm based on IAVL is proposed. In this scrambling, not only the rows and columns perform the circular motion, but also the exchange operation between the columns, which greatly improves the scrambling effect. Then, we designed a complex dynamic DNA coding technology. On this basis, the intra-block and inter-block diffusion algorithm is introduced. In the block, DNA addition, subtraction and XOR operations are dynamically selected to change the value of the pixel; the horizontal and vertical addition operations are implemented between blocks to achieve the diffusion effect. Experimental test results and security analyses are fully vindicated that our encryption scheme shows certain performance advantages in the face of classic performance tests and is easy to implement. Therefore, it is feasible in the field of digital communication.

AUTHOR CONTRIBUTIONS

Yuwen Sha carried out experiments, data analyzed and manuscript wrote. Yinghong Cao and Jun Mou made the theoretical guidance for this article. Huizhen Yan and Xinyu Gao improved the algorithm. All authors reviewed the manuscript.

CONFLICTS OF INTEREST

No conflicts of interests about the publication by all authors.

REFERENCES

[1] G. Ye, "A block image encryption algorithm based on wave transmission and chaotic systems," *Nonlinear Dyn.*, vol. 75, no. 3, pp. 417–427, Feb. 2014, doi: 10.1007/s11071-013-1074-6.

[2] S. Patel, V. Thanikaiselvan, D. Pelusi, B. Nagaraj, R. Arunkumar, and R. Amirtharajan, "Colour image encryption based on customized neural network and DNA encoding," *Neural Comput. Appl.*, pp. 1–18, May 2021, doi: 10.1007/s00521-021-06096-2.

[3] H. Hu, Y. Cao, J. Xu, C. Ma, and H. Yan, "An image compression and encryption algorithm based on the fractional-order simplest chaotic circuit," *IEEE Access*, vol. 9, pp. 22141–22155, 2021, doi: 10.1109/ACCESS.2021.3054842.

[4] 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, 2020, doi: 10.1049/iet-ipr.2018.5654.

[5] 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, doi: 10.1016/j.optlaseng.2020.106031.

[6] J. S. Khan and S. K. Kayhan, "Chaos and compressive sensing based novel image encryption scheme," *J. Inf. Secur. Appl.*, vol. 58, May 2021, Art. no. 102711, doi: 10.1016/j.jisa.2020.102711.

[7] X. Ma, J. Mou, J. Liu, C. Ma, and X. Zhao, "A novel simple chaotic circuit based on memristor–memcapacitor," *Nonlinear Dyn.*, vol. 100, no. 5, pp. 2859–2876, 2020, doi: 10.1007/s11071-020-05601-x.

[8] X. Li, J. Mou, L. Xiong, Z. Wang, and J. Xu, "Fractional-order double-ring erbium-doped fiber laser chaotic system and its application on image encryption," *Opt. Laser Technol.*, vol. 140, Aug. 2021, Art. no. 107074, doi: 10.1016/j.optlastec.2021.107074.

[9] S. S. Maniccam and N. G. Bourbakis, "Image and video encryption using SCAN patterns," *Pattern Recognit.*, vol. 37, no. 4, pp. 725–737, Apr. 2004, doi: 10.1016/j.patcog.2003.08.011.

[10] R.-J. Chen and S.-J. Horng, "Novel SCAN-CA-based image security system using SCAN and 2-D von neumann cellular automata," *Signal Process., Image Commun.*, vol. 25, no. 6, pp. 413–426, Jul. 2010, doi: 10.1016/j.image.2010.03.002.

[11] V. Chandee, C. David, D. Koukoulopoulos, and E. Smith, "The frequency of elliptic curve groups over prime finite fields," *Can. J. Math.*, vol. 68, no. 4, pp. 721–761, Aug. 2016, doi: 10.4153/CJM-2015-013-1.

[12] G. A. Morris and R. Freeman, "Selective excitation in Fourier transform nuclear magnetic resonance," *J. Magn. Reson.*, vol. 213, no. 2, pp. 214–243, Dec. 2011, doi: 10.1016/j.jmr.2011.08.031.

[13] X. Wang and D. Luan, "A novel image encryption algorithm using chaos and reversible cellular automata," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 18, no. 11, pp. 3075–3085, Nov. 2013, doi: 10.1016/j.cnsns.2013.04.008.

[14] F. Argoul, A. Arneodo, J. Elezgaray, G. Grasseau, and R. Murenzi, "Wavelet transform of fractal aggregates," *Phys. Lett. A*, vol. 135, nos. 6–7, pp. 327–336, 2017, doi: 10.1016/0375-9601(89)90003-0.

[15] J. Fridrich, "Symmetric ciphers based on two-dimensional chaotic maps," *Int. J. Bifurcation Chaos*, vol. 08, no. 06, pp. 1259–1284, Jun. 1998, doi: 10.1142/S021812749800098X.

[16] X. Chai, Z. Gan, Y. Lu, Y. Chen, and D. Han, "A novel image encryption algorithm based on the chaotic system and DNA computing," *Int. J. Mod. Phys. C*, vol. 28, no. 05, May 2017, Art. no. 1750069, doi: 10.1142/S0129183117500693.

[17] E. G. Nepomuceno, J. Arias-Garcia, L. G. Nardo, D. N. Butusov, and A. V. Tutueva, "Image encryption based on the pseudo-orbits from 1D chaotic map," *Chaos: Interdiscipl. J. Nonlinear Sci.*, vol. 29, no. 6, p. 61101, 2019, doi: 10.1063/1.5099261.

[18] 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, doi: 10.1016/j.sigpro.2017.07.034.

[19] A. Kulsoom, D. X. Aqeel-ur-Rehman, and S. A. Abbas, "An efficient and noise resistive selective image encryption scheme for gray images based on chaotic maps and DNA complementary rules," *Multimedia Tools Appl.*, vol. 75, no. 1, pp. 1–23, Jan. 2016, doi: 10.1007/s11042-014-2221-x.

[20] S. Zhu and C. Zhu, "Plaintext-related image encryption algorithm based on block structure and five-dimensional chaotic map," *IEEE Access*, vol. 7, pp. 147106–147118, 2019, doi: 10.1109/ACCESS.2019.2946208.

[21] J.-X. Chen, Z.-L. Zhu, C. Fu, H. Yu, and L.-B. Zhang, "A fast chaos-based image encryption scheme with a dynamic state variables selection mechanism," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 20, no. 3, pp. 846–860, Mar. 2015, doi: 10.1016/j.cnsns.2014.06.032.

[22] K. A. K. Patro and B. Acharya, "Secure multi–level permutation operation based multiple colour image encryption," *J. Inf. Secur. Appl.*, vol. 40, pp. 111–133, Jun. 2018, doi: 10.1016/j.jisa.2018.03.006.

[23] K. A. K. Patro and B. Acharya, "Novel data encryption scheme using DNA computing," in *Advances of DNA Computing in Cryptography*. Boca Raton, FL, USA: Taylor & Francis, 2018, pp. 69–110.

[24] X. Ouyang, Y. Luo, J. Liu, L. Cao, and Y. Liu, "A color image encryption method based on memristive hyperchaotic system and DNA encryption," *Int. J. Mod. Phys. B*, vol. 34, no. 04, Feb. 2020, Art. no. 2050014, doi: 10.1142/S0217979220500149.

[25] C. Chen, K. Sun, and S. He, "An improved image encryption algorithm with finite computing precision," *Signal Process.*, vol. 168, Mar. 2020, Art. no. 107340, doi: 10.1016/j.sigpro.2019.107340.

[26] K. A. K. Patro and B. Acharya, "An efficient dual-layer cross-coupled chaotic map security-based multi-image encryption system," *Nonlinear Dyn.*, vol. 104, no. 3, pp. 2759–2805, 2021, doi: 10.1007/s11071-021-06409-z.

[27] X. Liao, M. A. Hahsmi, and R. Haider, "An efficient mixed inter-intra pixels substitution at 2bits-level for image encryption technique using DNA and chaos," *Optik-Int. J. Light Electron Opt.*, vol. 153, pp. 117–134, Jan. 2018, doi: 10.1016/j.ijleo.2017.09.099.

[28] K. A. K. Patro, B. Acharya, and V. Nath, "Secure multilevel permutation-diffusion based image encryption using chaotic and hyper-chaotic maps," *Microsyst. Technol.*, vol. 25, no. 12, pp. 4593–4607, Dec. 2019, doi: 10.1007/s00542-019-04395-2.

[29] K. A. K. Patro, B. Acharya, and V. Nath, "A secure multi-stage one-round bit-plane permutation operation based chaotic image encryption," *Microsyst. Technol.*, vol. 25, no. 6, pp. 2331–2338, Jun. 2019, doi: 10.1007/s00542-018-4121-x.

[30] K. A. K. Patro, B. Acharya, and V. Nath, "Various dimensional colour image encryption based on non-overlapping block-level diffusion operation," *Microsyst. Technol.*, vol. 26, no. 5, pp. 1437–1448, May 2020, doi: 10.1007/s00542-019-04676-w.

[31] J. S. Khan and J. Ahmad, "Chaos based efficient selective image encryption," *Multidimensional Syst. Signal Process.*, vol. 30, no. 2, pp. 943–961, Apr. 2019, doi: 10.1007/s11045-018-0589-x.

[32] J. S. Khan, W. Boulila, J. Ahmad, S. Rubaiee, A. U. Rehman, R. Alroobaea, and W. J. Buchanan, "DNA and plaintext dependent chaotic visual selective image encryption," *IEEE Access*, vol. 8, pp. 159732–159744, 2020, doi: 10.1109/ACCESS.2020.3020917.

[33] K. Aditya, A. K. Mohanty, G. A. Ragav, V. Thanikaiselvan, and R. Amirtharajan, "Image encryption using dynamic DNA encoding and pixel scrambling using composite chaotic maps," in *Proc. IOP Conf., Mater. Sci. Eng.*, 2020, vol. 872, no. 1, Art. no. 012045, doi: 10.1088/1757-899X/872/1/012045.

[34] K. Patro, B. Acharya, and V. Nath, "Secure, lossless, and noise-resistive image encryption using chaos, hyper-chaos, and DNA sequence operation," *IETE Tech. Rev.*, vol. 37, no. 3, pp. 223–245, 2019, doi: 10.1080/02564602.2019.1595751.

[35] A. Rehman, X. Liao, A. Kulsoom, and S. A. Abbas, "Selective encryption for gray images based on chaos and DNA complementary rules," *Multimedia Tools Appl.*, vol. 74, no. 13, pp. 4655–4677, Jul. 2015, doi: 10.1007/s11042-013-1828-7.

[36] 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, doi: 10.1016/j.optlaseng.2016.08.009.

[37] K. A. K. Patro, A. Soni, P. K. Netam, and B. Acharya, "Multiple grayscale image encryption using cross-coupled chaotic maps," *J. Inf. Secur. Appl.*, vol. 52, Jun. 2020, Art. no. 102470, doi: 10.1016/j.jisa.2020.102470.

[38] H. R. Shakir, "A color-image encryption scheme using a 2D chaotic system and DNA coding," *Adv. Multimedia*, vol. 2019, Nov. 2019, Art. no. 7074264, doi: 10.1155/2019/7074264.

[39] X. Wang, Q. Wang, and Y. Zhang, "A fast image algorithm based on rows and columns switch," *Nonlinear Dyn.*, vol. 79, no. 2, pp. 1141–1149, Jan. 2015, doi: 10.1007/s11071-014-1729-y.

[40] Y. Zhang, B. Xu, and N. Zhou, "A novel image compression–encryption hybrid algorithm based on the analysis sparse representation," *Opt. Commun.*, vol. 392, pp. 223–233, Jun. 2017, doi: 10.1016/j.optcom.2017.01.061.

[41] 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, doi: 10.1016/j.ijleo.2014.06.054.

[42] Q. Yin and C. Wang, "A new chaotic image encryption scheme using breadth-first search and dynamic diffusion," *Int. J. Bifurcation Chaos*, vol. 28, no. 4, Apr. 2018, Art. no. 1850047, doi: 10.1142/S0218127418500475.

[43] 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, doi: 10.1016/j.sigpro.2016.12.008.

[44] 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, doi: 10.1016/j.compeleceng.2012.02.007.

[45] F. Yang, J. Mou, C. Luo, and Y. Cao, "An improved color image encryption scheme and cryptanalysis based on a hyperchaotic sequence," *Phys. Scripta*, vol. 94, no. 8, 2019, Art. no. 085206, doi: 10.1088/1402-4896/ab0033.

[46] A. N. Kengnou Telem, H. B. Fotsin, and J. Kengne, "Image encryption algorithm based on dynamic DNA coding operations and 3D chaotic systems," *Multimedia Tools Appl.*, vol. 80, no. 12, pp. 19011–19041, May 2021, doi: 10.1007/s11042-021-10549-0.

[47] M. Roy, S. Chakraborty, K. Mali, D. Roy, and S. Chatterjee, "A robust image encryption framework based on DNA computing and chaotic environment," *Microsyst. Technol.*, pp. 1–11, Jan. 2021, doi: 10.1007/s00542-020-05120-0.
[48] K. A. K. Patro, M. P. J. Babu, K. P. Kumar, and B. Acharya, "Dual-layer DNA-encoding–decoding operation based image encryption using one-dimensional chaotic map," in *Advances in Data and Information Sciences*, vol. 94. Singapore: Springer, 2020, pp. 67–80, doi: 10.1007/978-981-15-0694-9_8.
[49] Y. Liu, Z. Qin, X. Liao, and J. Wu, "A chaotic image encryption scheme based on Hénon–Chebyshev modulation map and genetic operations," *Int. J. Bifurcation Chaos*, vol. 30, no. 6, 2020, Art. no. 2050090, doi: 10.1142/S021812742050090X.
[50] X. An and S. Qiao, "The hidden, period-adding, mixed-mode oscillations and control in a HR neuron under electromagnetic induction," *Chaos, Solitons Fractals*, vol. 143, Feb. 2021, Art. no. 110587, doi: 10.1016/j.chaos.2020.110587.
[51] A. Xin-Lei, Q. Shuai, and Z. Li, "Dynamic response and control of neuros based on electromagnetic field theory," *Acta Phys. Sinica*, vol. 70, no. 5, 2021, Art. no. 050501.
[52] R. Ponuma and R. Amutha, "Compressive sensing based image compression-encryption using novel 1D-chaotic map," *Multimedia Tools Appl.*, vol. 77, no. 15, pp. 19209–19234, Aug. 2018, doi: 10.1007/s11042-017-5378-2.
[53] F. Ozkaynak, A. B. Ozer, and S. Yavuz, "Security analysis of an image encryption algorithm based on chaos and DNA encoding," in *Proc. 21st Signal Process. Commun. Appl. Conf. (SIU)*, Apr. 2013, pp. 1–4.
[54] R. Sivaraman, S. Rajagopalan, J. Bosco, 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 2020, doi: 10.1049/iet-ipr.2019.0618.
[55] J. S. Khan, J. Ahmad, S. S. Ahmed, H. A. Siddiqa, S. F. Abbasi, and S. K. Kayhan, "DNA key based visual chaotic image encryption," *J. Intell. Fuzzy Syst.*, vol. 37, no. 2, pp. 2549–2561, Sep. 2019, doi: 10.3233/JIFS-182778.
[56] C. Ma, J. Mou, P. Li, and T. Liu, "Dynamic analysis of a new two-dimensional map in three forms: Integer-order, fractional-order and improper fractional-order," *Eur. Phys. J. Special Topics*, Jun. 2021, doi: 10.1140/epjs/s11734-021-00133-w.
[57] T. Liu, S. Banerjee, H. Yan, and J. Mou, "Dynamical analysis of the improper fractional-order 2D-SCLMM and its DSP implementation," *Eur. Phys. J. Plus*, vol. 136, no. 5, p. 506, May 2021, doi: 10.1140/epjp/s13360-021-01503-y.
[58] T. Liu, H. Yan, S. Banerjee, and J. Mou, "A fractional-order chaotic system with hidden attractor and self-excited attractor and its DSP implementation," *Chaos, Solitons Fractals*, vol. 145, Apr. 2021, Art. no. 110791, doi: 10.1016/j.chaos.2021.110791.

YUWEN SHA received the B.S. degree from Dalian Polytechnic University, Dalian, China, in 2020, where he is currently pursuing the Ph.D. degree in control science and engineering. His research interests include chaos theory and chaotic digital image cryptosystems.

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.

HUIZHEN YAN received the B.S. and M.S. degrees from Xian 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.

XINYU GAO received the B.S. degree from Dalian Polytechnic University, Dalian, China, in 2020, where she is currently pursuing the Ph.D. degree in control science and engineering. Her research interests include chaos theory and chaotic digital image cryptosystems.

JUN MOU received the B.S., M.S., and Ph.D. degrees in physics and electronics from Central South University, Changsha, China. He is currently an Associate Professor with the School of Information Science and Engineering, Dalian Polytechnic University, China. His main research interests include nonlinear system control, secure communication, power system automation, and smart grid research.