A Novel Image Encryption Algorithm Based on Parameter-Control Scroll Chaotic Attractors

TAO WANG1,2, LIWEN SONG3, MINGHUI WANG1, AND ZHIBEN ZHUANG3

1 College of Computer Science, Sichuan University, Chengdu 610000, China
2 College of Forestry and Horticulture, Hubei Minzu University, Enshi 445000, China
3 School of Science, Hubei Minzu University, Enshi 445000, China

Corresponding author: Zhiben Zhuang (1697254891@qq.com)

This work was supported by the Special Funds for Double First-Class Construction in Hubei Province.

ABSTRACT

This paper describes a novel image encryption algorithm based on a parameter-control scroll chaotic attractor. Firstly, a novel chaotic system is constructed, and some basic dynamic characteristics of the chaotic system are analyzed. Secondly, the chaotic sequence is decomposed using the Schur decomposition method to obtain N orthogonal matrices, which are multiplied by the plaintext matrix. Finally, the six high-order bit planes are subjected to disturbance and “same OR” operations, and the scrambled image matrix is combined with the low two-bit matrix to obtain an eight-bit matrix. At the same time, the same OR operation is applied to the disturbed image matrix and the chaotic sequence to obtain the final encrypted image. Theoretical analysis shows that the key space of the algorithm can be up to 10135. Additionally, simulation results demonstrate that the proposed method achieves high key sensitivity and can effectively resist statistical analysis and gray value analysis attacks. With the exception of exhaustive attacks, all attack methods based on the determined plaintext–ciphertext mapping relationship will be invalid.

INDEX TERMS

Four-dimensional chaotic system, digital image encryption, dynamic characteristics, “same OR” operation.

I. INTRODUCTION

Rapid developments in Internet technologies have led to increased attention on the information security of digital images. To protect image information from illegal copying, use, and manipulation, experts and scholars have proposed various encryption algorithms such as chaotic sequences to disturb image bits [1], [2], chaotic sequences to perform XOR operations on pixel values of images [3], and the scrambling of pixels [1], [4]–[6].

In this paper, a novel image encryption algorithm based on a parameter control scroll chaotic attractor is proposed. Various chaotic equations with multi-scroll attractors have been reported [7]–[9]. Among them, the novelty of the multi-scroll system proposed by Karthikeyan et al. [7] depends on the number of scrolls determined by the control parameters, rather than by changing a discontinuous function. A new chaotic image scheme based on a breadth-first search and dynamic diffusion has been developed [10], and a four-dimensional multi-wing hyperchaotic attractor for use in image encryption has been proposed [11]. These systems can generate chaotic and hyperchaotic attractors under different parameters. The four-dimensional multi-wing hyperchaotic system is constructed by introducing a linear feedback device u based on a three-dimensional autonomous system. In this operation, the newly added controller u is fed back to the original controller y, and the original controller y is fed back to the new controller u at the same time. The four controllers of this system interact with each other through two operations, making the relationship somewhat complicated. The literature also describes a hyperchaotic multi-wing attractor generated in a 4D memristive circuit [12] as well as four-wing hyperchaotic attractors and two-, three-, and four-wing chaotic attractors generated in 4D memristive systems [13]. Additionally, the design and implementation of hyperchaotic circuits based on linear chirped magnetic field control memristors [14] and the construction, analysis, and improvement of hyperchaotic systems [15] have been reported. In 2017, Liu et al. [16] proposed an image based on chaos theory and the bit disturb operation was proposed in [17], and image encryption algorithms based on an orthogonal matrix and a chaotic map have been reported [18], [19]. Among them,
Ahmad et al. [18] proposed a compression sensing and noise-tolerant image encryption scheme based on chaotic maps and orthogonal matrices. The random orthogonal matrix generated in this system is directly multiplied with the plaintext image \( P \) to change the pixel value. Zhang et al. [20] proposed a new image scheme based on the spatiotemporal chaos of the Mixed Linear–Nonlinear Coupled Map Lattices (MLNCML). This scheme employed the strategy of DNA computing and one-time pad encryption policy. Zhang and Wang [21] proposed investigate the spatiotemporal chaos of the Mixed Linear–Nonlinear Coupled Map Lattices (MLNCML). This scheme employed the strategy of DNA computing and one-time pad encryption policy.

The rest of the paper is organized as follows. In Section 2, the chaotic system model is given. In Section 3, related knowledge is introduced. The encryption scheme is described in Section 4. Simulation results and performance analyses are reported in Section 5. Finally, the conclusions are drawn in Section 6.

### II. CHAOTIC SYSTEM MODEL

In 2018, Karthikeyan et al. [7] proposed a parameter-controlled multi-scroll chaotic attractor, whose system equation is as follows:

\[
\begin{align*}
    f(x) &= \text{sgn}(x) + \text{sgn}(x + b) + \text{sgn}(x - b) \\
          &+ \text{sgn}(x + 2b) + \text{sgn}(x - 2b) \\
    \dot{x} &= g(y - ax + f(x)) - c, \\
    \dot{y} &= x - y + z, \\
    \dot{z} &= -dy. \\
\end{align*}
\]

On the basis of the system (1), we introduce a controller \( u \) and a nonlinear function \( g(y) \), whereby \( u \) feeds back to \( y \) at the same time as the original controller \( y \) feeds back to the controller \( u \). These two operations allow the four system controllers to interact with each other, making the relationship more complex. The newly constructed chaotic system equation is as follows:

\[
\begin{align*}
    f(x) &= \text{sgn}(x) + \text{sgn}(x + b) + \text{sgn}(x - b) + \text{sgn}(x + 2b) + \text{sgn}(x - 2b) \\
    g(y) &= \text{sgn}(y) + \text{sgn}(y + b) + \text{sgn}(y - b) \\
    \dot{x} &= g(y - ax + f(x)) - c, \\
    \dot{y} &= x - y + z, \\
    \dot{z} &= -dy, \\
    \dot{u} &= ey^2 - u + g(y). \\
\end{align*}
\]

where \( a, b, c, d, e, g \) are system parameters and \( c \) is the control parameter. For example, when \( a = 0.3, b = 7.06, c = 1.89, d = -26.2, e = 2, g = 11 \), a four-winding chaotic attractor is produced.

### A. ANALYSIS OF THE NUMBER OF CHAOTIC ATTRACTIONS IN A SCROLL

By assigning different values to the control parameter \( c \), system (2) generates different numbers of scroll chaotic attractors. The number of chaotic attractors of different scrolls and their corresponding parameters are listed in Table 1, and the plane phase diagram is shown in Fig. 1.

| Multi-scroll | Parameters | Fig. |
|--------------|------------|-----|
| 1-Scroll     | 0.3, 7.06, -26.2, 2, 11, 38 | 1a  |
| 2-Scroll     | 0.3, 7.06, -26.2, 2, 11, 0.1 | 1b  |
| 3-Scroll     | 0.3, 7.06, -26.2, 2, 11, 0.7 | 1c  |
| 4-Scroll     | 0.3, 7.06, -26.2, 2, 11, 1.89 | 1d  |

### B. TIME SERIES ANALYSIS

When \( a = 0.3, b = 7.06, c = 1.89, d = -26.2, e = 2, g = 11 \), the values of \( x, y, z, u \) change with time \( t \) as shown in Fig. 2.

### C. FOUR-WINDING CHAOTIC ATTRACTOR PHASE DIAGRAM

When \( a = 0.3, b = 7.06, c = 1.89, d = -26.2, e = 2, g = 11 \), the three-dimensional phase diagram of the four-winding chaotic attractor generated by system (2) is as shown in Fig. 3.
III. RELEVANT KNOWLEDGE

A. BIT PLANE OF THE IMAGE

Information is stored in computers as binary data, which are expressed by 0s and 1s. Thus, the binary sequence of the pixel values in a digital image is represented by 0s and 1s. Taking the binary number of the position of each pixel value in the digital image constitutes an image of the same size as the original. The resulting image is referred to as the bit plane of the image. The bit plane position and weight of the grayscale image are presented in Table 2.

As can be seen from Table 2, the grayscale image can be decomposed into eight bit planes. As the weight of each bit plane is different, the amount of image information contained in each bit plane is also different. The percentage of information contained in each bit plane image is calculated as follows:

$$I(i) = \frac{2^{i-1}}{255} \times 100\%, \quad i = 1, 2, 3, 4, 5, 6, 7, 8 \quad (3)$$

The calculation results are presented in Table 3; the experimental bit plane decomposition diagram of the Lena image is shown in Fig. 5.

From Table 3, we find that the amount of image information contained in the bit planes gradually increases from low to high (that is, from the first to the eighth bit from the right). The lower two bit planes (from the 1st to the 2nd bit),
account for only 1.177% of the original image information, whereas and the higher six bit planes (from the 3rd to the 8th) contain 98.823% of the original information.

It can also be seen from Fig. 5 that, from the 8th bit plane to the 3rd bit plane, the image sharpness becomes progressively worse, but some image information is still visible. This shows that these six bit planes contain most of the information about the plaintext image. The images in the 2nd and 1st bit planes are blurred, which indicates that these two bit planes contain very little information about the plaintext image. Therefore, different encryption algorithms can be designed according to the amount of image information contained in each bit plane.

B. SAME OR OPERATION
The “same OR” operation and the “exclusive OR” operation have the same effect. The “same OR” operation is defined as follows: when the input variables are the same, the output is 1, and when the input variables are different, the output is 0. The calculation results are presented in Table 4.

C. SOME BASIC PROPERTIES AND ARITHMETIC RULES OF MATRICES
Let $Q$ be an orthogonal matrix and $P$ be a general matrix that satisfies the multiplication condition with $Q$. Then:

i. The inverse of $Q$ exists and $Q^{-1} = Q^T$.

ii. $Q^{-1}Q = QQ^{-1} = QQ^T = E$, where $E$ is the identity matrix.

iii. If $A = QP$, then $P = Q^{-1}A = Q^TA$.

iv. If $Q_1, Q_2, \ldots, Q_n$ are orthogonal matrices with the same rank as $Q$, and $A_1 = Q_1Q_2 \ldots Q_nP$ then $P = Q_n^{-1}Q_{n-1}^{-1} \ldots Q_1^{-1}A_1$.

IV. DESCRIPTION OF IMAGE ENCRYPTION ALGORITHM BASED ON CHAOTIC SEQUENCE
The flow chart of the encryption is shown in Fig. 6.

A. ENCRYPTION ALGORITHM
Given a grayscale image $A$, the encryption steps are as follows:

Step 1: Input grayscale image $A$, initial value of chaotic system $y_0 = [0.1, 0.2, 0.5]$, and the step size $L = 0.01$, and find the total iteration time $T = (200 + P \times P) \times L$, where $P = \max(m, n)$.

Step 2: Call the ode45 function, iterate system (2), and generate four chaotic sequences.

Step 3: Treat the four chaotic sequences as follows:

$$A1(:,:,i) = \text{reshape}(y(201:end - 1, i), P, P), i = 1, 2, 3, 4$$

(4)
where \( y(201 : end - 1, i) \) is used to take the value of chaotic sequence \( y(i) \) from 201 to \( (200 + P \times P) \) and \( \text{reshape}(y(201 : end - 1, i), P, P) \) is used to convert the extracted values of \( P \times P \) into the matrix of \( P \times P \).

**Step4:** The four matrices \( P \times P \) are subjected to Schur decomposition to produce four orthogonal matrices \( U_1, U_2, U_3, U_4 \).

**Step5:** If \( P = m \), then \( A_2 = U_4 \times U_3 \times U_2 \times U_1 \times A \); if \( P = n \), then \( A_2 = A \times U_1 \times U_2 \times U_3 \times U_4 \).

**Step6:** Repeat steps 3–6 \( n1(n1 \geq 2) \) times.

**Step7:** Convert \( A_2 \) into matrix \( A_3 \), which has one row and \( m \times n \) columns; map all the elements to integers between 0 and 255 using the following mapping formula:

\[
A3(i) = \text{round}(\frac{A3(i) - (\min(A3)) - 1}{\max(A3) - (\min(A3)) - 1}) \times 255; \quad i1 = 1, 2, 3, \ldots, P \times P
\]  

**Step8:** The fourth chaotic sequence is treated as follows:

\[
B = \text{mod}(\text{round}(y(200 : (N + 200), 1) \times 10^{15}), 5) + 1
\]

\[
B1 = \text{mod}(\text{round}(y(200 : (N + 200), 2) \times 10^{15}, (N - 1)) + 1
\]

\[
B2 = \text{mod}(\text{round}(y(200 : (N + 200), 2) \times 10^{15}), 256) + 1
\]

\[
B3 = \text{mod}(\text{round}(y(200 : (N + 200), 3) \times 10^{15}), 256) + 1
\]

\[
B4 = \text{mod}(\text{round}(y(200 : (N + 200), 4) \times 10^{15}), 256) + 1
\]

\[
B5 = B2 + B3
\]

\[
B6(i2) = B3(i2) \oplus B4(i2), \quad i2 = 1, 2, 3, \ldots, m \times n
\]

where \( N = m \times n \).

**Step9:** Convert all elements of sequence \( B5 \) to integers between 0 and 255. If \( B5(i3) > 255 \), then \( B5(i3) = B5(i3) - 255 \); if \( B5(i3) \leq 255 \), then \( B5(i3) = B5(i3) \).

**Step10:** Convert \( A3 \) to a binary number to obtain matrix \( A4 \), and extract the binary numbers of the first six columns of \( A4 \) (that is, the upper six bits of each row) to form the matrix \( A5 \).

**Step11:** Each row of binary numbers is disturbed in \( A5 \) to obtain the matrix \( A6 \). The disturbance formula is as follows:

\[
A6(i4, :) = \text{circshift}(A5(i4, :) \times B(i4), 2), \quad i4 = 1, 2, 3, \ldots, m \times n
\]  

where \( A5(i4, :) \) denotes all columns of row \( i4 \) in matrix \( A5 \), and \( \text{circshift}(A, k, 2) \) denotes that all elements of the row vector \( A \) are moved \( k \) units clockwise.

**Step12:** All column elements in \( A6 \) are disturbed to give \( A7 \). The disturbance formula is as follows:

\[
A7(:, i5) = \text{circshift}(A6(:, i5), B1(i5), 1), \quad i5 = 1, 2, 3, 4, 5, 6
\]

where \( A7(:, i5) \) denotes all rows of column \( i5 \) in matrix \( A7 \) and \( \text{circshift}(A, k, 1) \) denotes that all elements of the column vector \( A \) are moved \( k \) units clockwise.

**Step13:** Convert \( B5 \) into the binary matrix \( A8 \), containing \( m \times n \) rows and eight columns, and all elements of the first six columns of \( A8 \) (that is, the upper six bits of each row) are applied to the same OR operation with \( A7 \) to obtain matrix \( A9 \). This matrix is merged with the lower two bits of \( A4 \) (that is, the last two columns) to give a new binary matrix \( A10 \) consisting of \( m \times n \) rows and eight columns. \( A10 \) is then converted to a decimal number.

**Step14:** Use \( A10 \) to perform a bitwise same OR operation with \( B6 \), resulting in the final encrypted image \( A11 \).

### B. DECRYPTION ALGORITHM DESCRIPTION

**Step1:** Input the encrypted image matrix \( A11 \), initial value of the chaotic system \( y_0 = [0.1, 0.2, 2, 0.5] \), and the step size \( L = 0.01 \), and find the total iteration time \( T = (200, +P \times P) \times L \) where \( P = \max(m, n) \).

**Step2:** The fourth chaotic sequence is treated as follows:

\[
B = \text{mod}(\text{round}(y(200 : (N + 200), 1) \times 10^{15}), 5) + 1
\]

\[
B1 = \text{mod}(\text{round}(y(200 : (N + 200), 2) \times 10^{15}, (N - 1)) + 1
\]

\[
B2 = \text{mod}(\text{round}(y(200 : (N + 200), 2) \times 10^{15}), 256) + 1
\]

\[
B3 = \text{mod}(\text{round}(y(200 : (N + 200), 3) \times 10^{15}), 256) + 1
\]

\[
B4 = \text{mod}(\text{round}(y(200 : (N + 200), 4) \times 10^{15}), 256) + 1
\]

\[
B5 = B2 + B3
\]

\[
B6(i2) = B3(i2) \oplus B4(i2), \quad i2 = 1, 2, 3, \ldots, m \times n
\]

where \( N = m \times n \).

**Step3:** A bitwise same OR operation is applied to \( A11 \) and \( B6 \) to form a new sequence \( D \).

**Step4:** Convert \( D \) into the binary matrix \( D1 \), and take the binary numbers of the first six columns (that is, the upper six bits of each row) of \( D1 \) to form the matrix \( D2 \).
Step5: Convert sequence $B_5$ into the binary matrix $D_3$, and take all the elements of the upper six bits from $D_3$ to produce matrix $D_4$.

Step6: Perform a bitwise same OR operation on the matrices $D_4$ and $D_2$, and then restore the rows and columns to obtain the matrix $D_5$.

Step7: Combine the lower two bits of $D_5$ and $A_{11}$ (that is, the last two columns) into a binary matrix $D_7$ containing $m \times n$ rows and eight columns.

Step8: The four chaotic sequences are treated as follows:

$$D_8(:, :, i) = \text{reshape}(y(201: \text{end} - 1, i), P, P), i = 1, 2, 3, 4$$

(10)

Step9: The four $P \times P$ matrices obtained are separately subjected to Schur decomposition to give four orthogonal matrices $U_1, U_2, U_3, U_4$.

Step10: Multiply $D_7$ by $U_1^{-1}, U_2^{-1}, U_3^{-1}, U_4^{-1}$ to obtain the decrypted image $A$.

V. EXPERIMENTAL RESULTS AND ANALYSIS

A. EXPERIMENTAL PLATFORM

Experiments were conducted using a Windows 7 (64 bit) PC with an Intel(R) Core(TM) i5-6500 CPU @ 3.70 GHz and 8 GB RAM. The encryption algorithm described in the previous section was implemented by writing a program in Matlab R2014a.

B. EXPERIMENTAL RESULTS

The experiment was performed using grayscale versions of the Lena, Boat, Baboon, Peppers, Couple and Leaf images. Each image contained $256 \times 256$ pixels, although the proposed method can be applied to grayscale images of other sizes. The plaintext, encrypted, and decrypted images are shown in Fig. 7.

C. KEY SPACE ANALYSIS

One of the most important factors determining the strength of an image encryption algorithm is the size of the key space. The key space is the digital space that can be used as an encryption and decryption key. A larger key space produces a stronger ability to resist brute force attacks. The key of the method described in this paper consists of four initial values $y_0$ and the system parameters $a, b, c, d, e, g$. With a computer accuracy of $10^{-15}$, the key space of the algorithm is $10^{135} > 2^{100}$ (if the key space of an image encryption algorithm is greater than $2^{100}$, then it is considered to be safe[22], [23]). Therefore, the proposed algorithm is sufficiently secure.

D. ANALYSIS OF INFORMATION ENTROPY

Information entropy is an important indicator of randomness, which reflects the distribution of gray values of images. If the gray value distribution is uniform, then the information entropy of the image is relatively large. The information entropy of an image is calculated as:

$$H(C) = -\sum_{i=1}^{L} p(x_i) \log_2 p(x_i)$$

(11)
where \( p(x_i) \) is the probability of \( x_i \) and \( L \) is the total number of \( x_i \). For grayscale images, the theoretical maximum information entropy is 8; the closer the image information entropy is to the theoretical maximum, the more random the grayscale distribution of image pixels. The information entropy of Lena, Baboon, Boat, Peppers, and Couple before and after encryption is presented in Table 5. The simulation results show that the pixel value distribution of the encrypted images is very uniform, and the algorithm has a good encryption effect.

Taking Lena as an example, the information entropy after encryption is compared with the information entropy of several methods described in the literature [16], [17], [24]–[26]. The comparative results in Table 6 demonstrate that the information entropy after image encryption by the proposed algorithm is obviously superior to that following encryption by the previous methods [16], [17], [24]–[26], and so the proposed algorithm has a better encryption effect.

**E. ANALYSIS OF FIXED-POINT RATIO AND GRAY AVERAGE CHANGE VALUE**

The fixed-point ratio is the percentage of pixel points whose gray value does not change after the image is encrypted; it is given by:

\[
BD(G, C) = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} f(i, j)}{MN} \times 100\% \quad (12)
\]

where \( f(i, j) = \begin{cases} 1, & g_y = c_y \\ 0, & g_y \neq c_y \end{cases} \). The fixed points of the algorithm as calculated by Eq. (12) are presented in Table 7.

The gray average change value evaluates the gray level change of the encrypted image. This is calculated as:

\[
GAVE(C, G) = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} |c_y - g_y|}{MN} \quad (13)
\]

where \( G \) is the plaintext image and \( C \) is the ciphertext image. The average change in the gray level produced by the proposed algorithm is presented in Table 8.

**F. HISTOGRAM ANALYSIS**

The image histogram reflects the distribution of image pixel values. A flatter histogram indicates that the pixel value distribution is more uniform. Figure 8 shows histograms of the original and encrypted versions of the Lena, Baboon, Boat, Peppers, and Couple images. The variances of histograms of an image with 256 gray levels can be computed by

\[
\text{var}(Z) = \frac{1}{256^2} \sum_{i=0}^{255} \sum_{j=0}^{255} (z_i - z_j)^2 \quad (14)
\]

where \( Z = \{z_0, z_1, \ldots, z_{255}\} \) is the vector of the histogram values, \( z_i \) and \( z_j \) are the numbers of pixels which gray values are equal to \( i \) and \( j \), respectively.

Table 9 lists the variances of the histogram of the plain image and cipher image. When the variance is lower, the uniformity of the image is higher [30]. The variances of the cipher images are largely reduced compared to those of the plain image in Table 9.

From Table 9, we can see that the cipher images generated by our encryption algorithm are random enough and the proposed encryption scheme has very high security level.

**G. KEY SENSITIVITY ANALYSIS**

A small change in the key will result in a great change in ciphertext. This is the key sensitivity. This experiment used the Boat image as an example. Figure 9 shows the sensitivity of the algorithm to the initial key. Figure 9(a) is
FIGURE 9. Key sensitivity experiment analysis (a) Plaintext image, (b) Ciphertext $Y_0$ (key is $y_0$), (c) Ciphertext $Y_1$ (key is $y_1$), (d) Correct $Y_0$ decryption result, (e) $Y_0$ using $y_1$: incorrect decryption result, (f) $Y_1$ using $y_0$: incorrect decryption result.

The difference between the two images can also be measured by the pixel change rate (NPCR) and the normalized mean change intensity (UACI). These metrics are calculated as follows:

$$\text{NPCR} = \frac{\sum_{i,j} D(i,j)}{M \times N} \times 100\%$$

(15)

$$\text{UACI} = \frac{\sum_{i,j} |C_1(i,j) - C_2(i,j)|}{M \times N \times 255} \times 100\%$$

(16)

where $D(i,j) = \begin{cases} 1, & C_1(i,j) \neq C_2(i,j) \\ 0, & C_1(i,j) = C_2(i,j) \end{cases}$, $M \times N$ is the image size, and $C_1(i,j), C_2(i,j)$ are the pixel values of the two images at position $(i,j)$. Larger values of NPCR and UACI indicate a greater difference between the two images. To better evaluate the key sensitivity of the algorithm,
TABLE 10. Key sensitivity test results.

| Images  | Lena | Baboon | Boat |
|---------|------|--------|------|
| Test value | 0.9961 | 0.3361 | 0.9965 | 0.3361 | 0.9957 | 0.335 |
| Ref. [29] | 0.9957 | 0.3508 | — | — | — | — |
| Ref. [30] | 0.9962 | 0.3344 | 0.9961 | 0.3349 | — | — |
| Ref. [31] | 0.9962 | 0.3347 | — | — | 0.996 | 0.334 |
| Ref. [32] | 0.9961 | 0.3347 | 0.9961 | 0.3347 | — | — |
| Ref. [33] | 0.9954 | 0.33443 | — | — | — | — |

TABLE 11. Test results of correlation coefficient between plaintext and ciphertext images.

| Image   | Horizontal correlation coefficient | Vertical correlation coefficient | Diagonal direction correlation coefficient |
|---------|-----------------------------------|----------------------------------|------------------------------------------|
| Lena    | 0.949                             | -0.0327                          | 0.907                                    |
| Baboon  | 0.6961                           | -0.0138                          | 0.829                                    |
| Boat    | 0.9625                           | -0.0031                          | 0.8296                                  |
| Peppers | 0.979                            | -0.0036                          | 0.9706                                  |
| Cowpel | 0.8881                           | 0.0082                           | 0.9195                                  |

Next experiment used the Boat image as an example. Figure 10 shows the sensitivity of the algorithm to the initial key. Figure 10(a) is the plaintext image, Figs. 10(b) and 10(c) show the encrypted ciphertext images \( W_0 \) and \( W_1 \) given by the keys \( w_0 = [0.1, 0.2, 2, 0.5] \) and \( w_1 = [0.100000000000001, 0.2, 2, 0.5] \). The test values given by several previous methods [29]–[33] are presented in Table 10. It can be seen that the proposed algorithm has a good test effect, demonstrating that the encryption algorithm proposed in this paper has good key sensitivity.

H. CORRELATION ANALYSIS OF ADJACENT PIXELS

One feature of digital images is the strong correlation between adjacent pixels. To calculate the correlation between adjacent pixels before and after encryption, 5000 sets of adjacent pixels were randomly selected in the horizontal, vertical, and diagonal directions of the plaintext and ciphertext images. The horizontal, vertical, and diagonal correlation coefficients were calculated using Eq. (17), and the test results are listed in Table 11. The pixel correlation of the Peppers plaintext image and the ciphertext image in the horizontal, vertical, and diagonal directions is shown in Fig. 11.

From Fig. 11, it is clear that the pixel points of the plaintext image in all three directions are concentrated near the diagonal, indicating strong correlation between the pixels of the

\[
\begin{align*}
E(x) &= \frac{1}{K} \sum_{i=1}^{K} x_i, \\
D(x) &= \frac{1}{K} \sum_{i=1}^{K} (x_i - E(x))^2, \\
Cov(x, y) &= \frac{1}{K} \sum_{i=1}^{K} (x_i - E(x))(y_i - E(y)), \\
r_{xy} &= \frac{Cov(x, y)}{\sqrt{D(x)D(y)}}
\end{align*}
\]

(17)
plaintext image. In contrast, the pixels are evenly distributed and scattered in all three directions of the ciphertext image, indicating that the correlation between the pixels has been destroyed. Therefore, the proposed algorithm can effectively reduce the strong correlation between adjacent pixels of the image and has a good effect on image encryption.

I. ANTI-SHEAR ATTACK ANALYSIS

To test the anti-shearing ability of the algorithm, we cut off the image of $40 \times 40$ in the middle of the encrypted image, as shown in Fig. 12(b), and then decrypted the cut image. The decrypted image is shown in Fig. 12(d). Fig. 12(a) shows the original encrypted image and Fig. 12(c) shows the decrypted image of the original encrypted image. Comparing Fig. 12(c) with Fig. 12(d), the pixel values of some points in Fig. 12(d) changed, but the general information about the plaintext image could still be displayed. Therefore, the encrypted image still had a certain decryption effect after being subjected to a cut attack.

J. ANTI-NOISE ATTACK ANALYSIS

Salt and pepper noise is a type of noise that is often seen in images. Salt and pepper noise may occur when image signals are suddenly and strongly interfered with, or result from analog digital converters or bit transmission errors. To test the anti-salt noise attack ability of the proposed algorithm, we added 10% salt and pepper noise to the Lena encrypted image. The encrypted image after noise is shown in Fig. 13(b). The noisy image was then decrypted, and the decrypted image is shown in Fig. 13(d). Fig. 13(a) shows the original encrypted image and Fig. 13(c) shows the original
decrypted image. Comparing Fig. 13(c) with Fig. 13(d), some pixel values in Fig. 13(d) changed, but the approximate information of the original plaintext image could still be displayed. This shows that the encrypted image still had a certain decryption effect after being attacked by salt and pepper noise.

VI. CONCLUSION
This paper has described a new image encryption algorithm based on a parameter-control scroll chaotic attractor. Firstly, a novel chaotic system was constructed, and some basic dynamic characteristics of the chaotic system were analyzed. Secondly, the chaotic sequence was decomposed using the Schur decomposition method to obtain N orthogonal matrices. These N orthogonal matrices were multiplied by the plaintext matrix. Finally, the six high-order bit planes were subjected to disturbance and “same OR” operations, and the scrambled image matrix was combined with the low two-bit matrix to obtain an eight-bit matrix. At the same time, the same OR operation was applied to the disturbed image matrix and the chaotic sequence to obtain the final encrypted image. Comprehensive analysis has demonstrated that the proposed algorithm achieves a large key space and has a good effect on image encryption.

REFERENCES
[1] S. Sun, “A novel hyperchaotic image encryption scheme based on DNA encoding, pixel-level scrambling and bit-level scrambling,” IEEE Photon. J., vol. 10, no. 2, pp. 1–14, Apr. 2018.
[2] L. Rui, “New algorithm for color image encryption using improved 1D chaotic logistic map,” Open Cybern. Syst. J., vol. 9, no. 1, pp. 210–216, Apr. 2015.
[3] X. Chai, Z. Gan, and M. Zhang, “A fast chaos-based image encryption scheme with a novel plain image-related swapping block permutation and block diffusion,” Multimedia Tools Appl., vol. 76, no. 14, pp. 15561–15585, Sep. 2016.
[4] T. Gao and Z. Chen, “Image encryption based on a new total shuffling algorithm,” Chaos, Solitons Fractals, vol. 38, no. 1, pp. 213–220, Oct. 2008.
[5] N. K. Pareek, V. Patidar, and K. K. Sud, “Image encryption using chaotic logistic map,” Image Vis. Comput., vol. 24, no. 9, pp. 926–934, Sep. 2006.
[6] J. Ahmad and S. O. Hwang, “Chaos-based diffusion for highly autocorrelated data in encryption algorithms,” Nonlinear Dyn., vol. 82, no. 4, pp. 1839–1850, Jul. 2015.
[7] X. Rajagopal, S. Çiçek, P. Nasereddinoussavi, A. J. M. Khalaf, S. Ifafari, and A. Karthikeyan, “A novel parametrically controlled multi-scroll chaotic attractor along with electronic circuit design,” Eur. Phys. J. Plus, vol. 133, no. 9, Sep. 2018.
[8] M. García-Martínez, L. J. Ontañón-García, E. Campos-Cantón, and S. Čelikovský, “Hyperchaotic encryption based on multi-scroll piecewise linear systems,” Appl. Math. Comput., vol. 270, pp. 413–424, Nov. 2015.
[9] P. Wang, Y. Feng, and N. X. Sun, “Chaotic phenomena in linear systems by using nonlinear feedback,” Control Theory Appl., vol. 21, no. 5, pp. 4–6, 2002.
[10] 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. 04, May 2018, Art. no. 1850047.
[11] Z. P. Feng, C. H. Wang, and Y. Lin, “A novel four-dimensional multi-wing hyper-chaotic attractor and its application in image encryption,” Acta Phys. Sinica, vol. 63, no. 24, 2014, Art. no. 240506.
[12] L. Zhou, C. Wang, and L. Zhou, “Generating hyperchaotic multi-wing attractor in a 4D memristive circuit,” Nonlinear Dyn., vol. 85, no. 4, pp. 2653–2663, May 2016.
[13] L. Zhou, C. Wang, and L. Zhou, “Generating four-wing hyperchaotic attractor and two-wing, three-wing, and four-wing chaotic attractors in 4D memristive system,” Int. J. Bifurcation Chaos, vol. 27, no. 02, Mar. 2017, Art. no. 1750027.
[14] C. Wang, L. Zhou, and R. Wu, “The design and realization of a hyper-chaotic circuit based on a flux-controlled memristor with linear memductance,” J. Circuits, Syst. Comput., vol. 27, no. 03, Oct. 2017, Art. no. 1850038.
[15] Y. Liu, Study on Chaos Based Pseudorandom Sequence Algorithm and Image Encryption Technique. Harbin, China: Harbin Institute of Technology, 2015.
[16] J. Liu, D. Yang, H. Zhou, and S. Chen, “A digital image encryption algorithm based on bit-planes and an improved logistic map,” Multimedia Tools Appl., vol. 77, no. 8, pp. 10217–10233, Nov. 2017.
[17] M. Hao, “Image encryption algorithm based on multiple Chaotic Systems and bit operations,” Res. Explor. Libr., vol. 34, no. 3, pp. 35–39, 2015.
[18] J. Ahmad, M. A. Khan, and S. O. Hwang, “A compression sensing and noise-tolerant image encryption scheme based on chaotic maps and orthogonal matrices,” Neural Comput. Appl., vol. 28, pp. 953–967, Dec. 2017.
[19] J. Ahmad, M. A. Khan, and F. Ahmed, “A novel image encryption scheme based on orthogonal matrix, skew tent map, and XOR operation,” Neural Comput. Appl., vol. 3, pp. 1–11, Dec. 2017.
[20] Y.-Q. Zhang, X.-Y. Wang, J. Liu, and Z.-L. Chi, “An image encryption scheme based on the MLNCMl system using DNA sequences,” Opt. Lasers Eng., vol. 82, pp. 95–103, Jul. 2016.
[21] Y.-Q. Zhang and X.-Y. Wang, “Spatiotemporal chaos in mixed linear–nonlinear coupled logistic map lattice,” Phys. A, Stat. Mech. Appl., vol. 402, pp. 104–118, May 2014.
[22] R. Enayatifar, A. H. Abdollah, I. F. Isnin, A. Altameem, and M. Lee, “Image encryption using a synchronous permutation-diffusion technique,” Opt. Lasers Eng., vol. 90, pp. 146–154, Mar. 2017.
[23] 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.
[24] B. Li, X. Liao, and Y. Jiang, “A novel image encryption scheme based on logistic map and dynamotic modular curve,” Multimedia Tools Appl., vol. 77, no. 7, pp. 8911–8938, May 2017.
[25] Y. Liu, T. Xj, and J. Ma, “Image encryption algorithm based on hyper-chaotic system and dynamic S-box,” Multimedia Tools Appl., vol. 75, no. 13, pp. 7739–7759, 2016.
[26] Y. Gd and H. Xi, “A novel block chaotic encryption scheme for remote sensing image,” Multimedia Tools Appl., 2016, vol. 75, no. 18, pp. 1–14.
[27] Z.-H. Gan, X.-L. Chai, D.-I. Han, and Y.-R. Chen, “A chaotic image encryption algorithm based on 3-D bit-plane permutation,” Neural Comput. Appl., vol. 31, no. 11, pp. 7111–7130, May 2018.
[28] 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.
[29] Y. Abanda and A. Tiedeu, “Image encryption by chaos mixing,” IET Image Process., vol. 10, no. 10, pp. 742–750, Oct. 2016.
[30] Y. He, Y.-Q. Zhang, and X.-Y. Wang, “A new image encryption algorithm based on two-dimensional spatiotemporal chaotic system,” Neural Comput. Appl., vol. 32, no. 1, pp. 247–260, Aug. 2018.
[31] G. Ye, C. Pan, X. Huang, and Q. Mei, “An efficient pixel-level chaotic image encryption algorithm,” Nonlinear Dyn., vol. 94, no. 1, pp. 745–756, Jun. 2018.
[32] Y. Zhang, “The unified image encryption algorithm based on chaos and cubic S-Box,” Inf. Sci., vol. 450, pp. 361–377, Jun. 2018.
[33] M. Brindha and N. A. Gounden, “A chaos based image encryption and lossless compression algorithm using hash table and Chinese remainder theorem,” Appl. Soft Comput., vol. 40, pp. 379–390, Mar. 2016.

TAO WANG received the M.Sc. degree in computer science from the Huazhong University of Science and Technology, in 2005. He is currently pursuing the Ph.D. degree in software engineering with the College of Computer Science, Sichuan University. He is currently an Associate Professor with the School of Advanced Materials and Mechatronic Engineering, Hubei Minzu University. He is also the author of more than ten research articles and a series of computer software product registration. His research interests include data mining, image encryption, and parallel computing.
LIWEN SONG received the B.E. degree in information and computing science from Hubei Minzu University, China, where he is currently pursuing the M.S. degree in operational research and cybernetics with the School of Science. His research interests mainly focus on digital image processing and JAVA programming.

MINGHUI WANG was born in 1971. He is currently a Professor with the College of Computer Science, Sichuan University, Chengdu, China. His current research interests include information fusion, image processing, intelligent transportation systems, and multiple target tracking.

ZHIBEN ZHUANG received the B.Sc. degree, in 2015. He is currently pursuing the M.S. degree in operational research and cybernetics with the School of Science, Hubei Minzu University, China. His research interest includes digital image encryption.