The Visual Computer (2023) 39:4955–4983
https://doi.org/10.1007/s00371-022-02640-w

ORIGINAL ARTICLE

A novel fast image encryption scheme based on a new one-dimensional compound sine chaotic system

Jianeng Tang[1] · Feng Zhang[2] · Hui Ni[2]

Accepted: 20 July 2022 / Published online: 28 August 2022

Abstract

In the paper, a new one-dimensional (1-D) compound Sine chaotic system (CSCS) is first proposed. Then new chaotic maps are generated by the CSCS. And four novel generated maps are used for the illustration about the CSCS. Secondly, the results of performance analysis show that the four maps have large Lyapunov exponents and high complexity. Thirdly, a high-efficiency image encryption scheme is proposed by employing two of the four new produced chaotic maps. In the proposed encryption scheme, the simplest diffusion operation is used. And we use a variety of scrambling operations, such as Zigzag transform, Magic confusion and the row confusion. In addition, to increase key space and in order to improve the ability to resist two kinds of attacks, namely the known plaintext attack and the selected plaintext attack, the control parameters and the initial values of the two new chaotic systems are generated based on the SHA-256 function. Finally, compared to other schemes, simulation tests show that our scheme not only has higher security but also faster encryption speed.

Keywords Compound chaotic system · Chaotic maps · Image encryption · SHA-256 function · Security analysis

1 Introduction

In the era of 5G network coverage, a large number of digital data are spread based on a variety of digital terminals, such as 5G mobile phones, computers, iPads and wearable devices [1,2]. At present, there are some major threats to digital content data, such as various computer viruses and hacker attacks. Therefore, it becomes a significant problem about the protection and safe use of digital content data. In these numeral content data, one of the typical data is digital image. In addition, digital image transmission requires a strong real-time property in the communication. But traditional encryption technologies are found to be inefficient for these digital images [3], such as Data Encryption Standard and Advanced Encryption Standard. Therefore, researchers have proposed many different image encryption techniques, including one-time keys [4], DNA coding [5,6], spatial bit-level permutation [7], perceptron model [8], dynamic random growth technique [9]. However, among these technologies, because of many similar properties between chaotic systems and cryptosystems [10–12], image encryption schemes using chaotic maps have been widely studied [13–17]. In recent years, researchers have used some new methods to design image encryption algorithms, such as parallel computing system [18], piecewise coupled map lattice [19], matrix semi-tensor product theory [20,21], fractal sorting matrix [22,23], compressed sensing [17,24].

Generating the security key is an important part of an encryption system. We can use 1-D chaos systems and multi-dimensional(MD) chaos systems to generate the security key. At present, according to MD chaos systems, image encryption schemes are widely used [25–28]. A novel image encryption algorithm was proposed according to a 2-D compound homogeneous hyper-chaotic system [29]. However, their hardware implementations become more difficult because of the complex structures and many parameters of the MD chaos systems. At the same time, 1-D chaos systems have some merits: the simple structure, the easy implementation and lower computation-cost.

✉ Jianeng Tang
tangjianeng@sina.com

Feng Zhang
fzhang_mm@163.com

Hui Ni
hni1234mm@88.com

[1] College of Engineering, Huaqiao University, Quanzhou 362021, China

[2] Fujian MM Electronics Co., Ltd., Quanzhou 362000, China

Recently, by use of 1-D chaos maps, researchers have put forward a lot of image encryption algorithms [3,13,30–32]. But 1-D chaotic maps may exhibit drawbacks [33,34]: (1) their chaotic intervals are finite; (2) their output states are non-uniform. Liu *et al.* analyzed two image encryption algorithms according to a first-order time-delay system [35]. Some image encryption schemes may be destroyed because of these defects [36,37]. Thus, it is very important to study a new 1-D chaos system with good chaotic properties.

To address these above issues, a new 1-D chaos system was put forward by use of two common 1-D chaos maps [3]. Similarly, Hua *et al.* [13] put forward a cosine-transform-based chaotic system (CTBCS) to generate novel chaotic maps. Using the CTBCS, they produced three novel chaotic maps. We can observe that the Lyapunov exponents(LEs) with the three new 1-D chaos systems are maintained at about 1.4. Combining the Logistic map and the Cubic map [32], we can generate the Logistic-Cubic-Cosine (LCC) map by use of the CTBCS. However, we can see that the whole data range cannot be randomly covered by the output of the LCC mapping. This means that the output of the LCC mapping may not have the characteristic of high randomness. At the same time, this paper [13] does not point out the measures to improve the performance of the chaotic system. Therefore, a new 1-D compound Sine chaotic system (CSCS) is proposed with more complex behaviors. Based on two 1-D chaos systems, a novel chaos map can be produced by the CSCS. And we give an effective method to make the output of the novel chaos map randomly covered in the whole range. To obtain larger LEs, our method is simpler than that in Ref. 38. Here we summarize the innovations of this paper, as shown below.
(1) We propose a new 1-D compound Sine chaotic system (CSCS). By utilizing two common 1-D chaos maps, a lot of novel chaos systems with excellent performance can be generated by the CSCS. Then, we produce four novel 1-D chaos maps to confirm the practicability of the CSCS.
(2) We do the performance analyses of four new chaotic maps. A novel method is proposed to make these new mappings obtain larger Lyapunov exponents. And simulations indicate that these maps generated by the CSCS exhibit complex chaos behaviors.
(3) Based on two maps produced by the CSCS, Zigzag transform and Magic confusion, we design a novel fast image encryption scheme.
(4) Experimental data show that our encryption scheme is faster and better than other encryption algorithms.

We organize the rest of the paper as follows. In Sect. 2, we present the CSCS. Then, four novel chaos maps are produced by the CSCS. And it is analyzed about the performances of the four new chaotic maps. Section 3 proposes a new image encryption scheme based on two maps produced by the CSCS, Zigzag transform and Magic confusion. We simulate the proposed algorithm by use of various types of grayscale images in Sect. 4. Then, it is compared with other encryption schemes. Finally, we draw the conclusions in Sect. 5.

2 CSCS

In this section, four existing chaotic maps are first introduced as seed maps. Next discrete cascade chaos [32] and CTBCS [13] are analyzed. Then, a new 1-D compound Sine chaotic system (CSCS) is proposed. Finally, these chaotic maps generated by the CSCS have best chaotic behaviors compared with discrete cascade chaos and CTBCS. Here, bifurcation diagrams, sample entropy [39], NIST SP800-22 [40] and LEs are used to evaluate its chaotic performance.

2.1 Common maps

In the section, four common maps are introduced, including Tent map, Cubic map, Sine map and Logistic map. Mathematically, the four existing common maps can be defined as

$$\begin{aligned}
&\textit{Tent map}: x_{n+1} = \mathcal{T}(\alpha, x_n) = 1 - 2\alpha|x_n - (2\alpha)^{-1}| \\
&\textit{Cubic map}: x_{n+1} = \mathcal{C}(\alpha, x_n) = |4x_n^3 - 3\alpha x_n| \\
&\textit{Sine map}: x_{n+1} = \mathcal{S}(\alpha, x_n) = \alpha \sin(\pi x_n) \\
&\textit{Logistic map}: x_{n+1} = \mathcal{L}(\alpha, x_n) = 4\alpha x_n(1 - x_n)
\end{aligned}$$

where $|\cdot|$ represents an absolute value operation and α is the control parameter, while $\alpha \in (0, 1)$.

2.2 Discrete cascade chaos

It is found that the cascade of chaotic systems can considerably improve the LEs of cascade chaos and increase system parameters and expand parameter regions of chaos mapping and full mapping [32]. We can use the following definition to illustrate discrete cascade chaos.

Definition 1 Given two different discrete chaotic seed maps $\mathcal{F}_1(a, x_n)$ and $\mathcal{F}_2(b, x_n)$, $x_n \in \mathbf{D}'$, $\mathcal{F}_1(a, x_n) \in \mathbf{D}_1$, $\mathcal{F}_2(b, x_n) \in \mathbf{D}_2$, $n = 0, 1, 2, \ldots$. If $\mathbf{D}' = \mathbf{D}_1 = \mathbf{D}_2$ is satisfied, we can get a new cascaded chaotic map. Discrete cascade chaos is defined as

$$x_{n+1} = \mathcal{F}_2(b, \mathcal{F}_1(a, x_n)) \tag{1}$$

Here the control parameters include a and b. And we set $\mathbf{D}' = [0, 1]$ in the paper.

By use of two of the four common maps, we can get twelve 1-D cascade chaos maps combining Eq. (1). And here we introduce four cascade chaotic maps as shown in Table 1. For example, we can make use of the Cubic map and the Logistic

Table 1 Four cascade chaotic maps

$\mathcal{F}_1(a, x_n)$	$\mathcal{F}_2(b, x_n)$	Cascade chaotic maps	Definition
$\mathcal{L}(\alpha, x_n)$	$\mathcal{S}(1, x_n)$	Logistic-Sine(LS)	$x_{n+1} = \sin(\pi 4\alpha x_n(1 - x_n))$
$\mathcal{S}(\alpha, x_n)$	$\mathcal{T}(1, x_n)$	Sine-Tent(ST)	$x_{n+1} = 1 - 2\lvert\alpha \sin(\pi x_n) - 0.5\rvert$
$\mathcal{L}(\alpha, x_n)$	$\mathcal{T}(1, x_n)$	Logistic-Tent(LT)	$x_{n+1} = 1 - 2\lvert 4\alpha x_n(1 - x_n) - 0.5\rvert$
$\mathcal{L}(\alpha, x_n)$	$\mathcal{C}(1, x_n)$	Logistic-Cubic(LC)	$x_{n+1} = \lvert 4o^3 - 3o\rvert, o = 4\alpha x_n(1 - x_n)$

map to produce the Logistic-Cubic(LC) map. Similarly, we can generate the LS map, the ST map and the LT map.

2.3 Common maps

Mathematically, we can use the following formula to describe the CTBCS.

$$x_{n+1} = \cos\{\pi[\mathcal{H}_1(c, x_n) + \mathcal{H}_2(d, x_n) + \gamma]\} \quad (2)$$

where $\mathcal{H}_1(c, x_n)$ and $\mathcal{H}_2(d, x_n)$ are two common chaos maps in Sect. 2.1. The control parameters include c and d, and the parameter γ is variable. In the paper [13], authors made use of the Logistic chaos map and the Sine chaos map to produce the Logistic-Sine-Cosine (LSC) chaotic map combining Eq. (2). Similarly, they produced the Sine-Tent-Cosine(STC) chaotic map. They also produced the Tent-Logistic-Cosine(TLC) chaos map. When $\gamma = 0.5$, the three maps have chaotic intervals. At the same time, the whole data range can be randomly covered by their outputs. This means that their outputs may have the characteristic of high randomness. When the control parameter α changes from 0 to 1, the LEs of the three chaotic maps are maintained at about 1.4.

According to the Logistic map and the Cubic map [32], we can generate the Logistic-Cubic-Cosine (LCC) map by use of the CTBCS. Mathematically, the LCC system can be represented as

$$x_{n+1} = \cos\{\pi[4\alpha x_n(1-x_n) + \lvert 4x_n^3 - 3(1-\alpha)x_n\rvert + \gamma]\} \quad (3)$$

where $c = \alpha, d = 1 - \alpha$.

Considering the periodicity of Cosine function, we set $\gamma \in [-1, 1]$. As shown in Fig. 1, we can see that the bifurcation diagrams of the LCC map are different from the above three chaos maps, namely, the LSC chaotic map, the STC chaotic map and the TLC chaos map. Figure 2 presents the LE distributions about the LCC map. From Figs. 1 and 2, when we change the shifting constant γ from -1 to 1, the whole data range cannot be randomly covered by the output of the LCC map. This means that the output of the LCC mapping may not have the characteristic of high randomness. That is, the chaotic performance of the LCC mapping is not as good as that of the three maps [13]. At the same time, there is no method to improve the performance of chaotic systems. A method was proposed and the LEs of Chebyshev–Chebyshev system were maintained at about 1.8 [3]. Some researchers increased the dimension of chaotic system to obtain larger LE [38]. The study show that the system can theoretically generate larger positive LE as long as the dimension of system is sufficiently high. With consideration of these issues, we propose a new 1-D compound Sine chaotic system (CSCS) to produce chaos maps with excellent performance.

2.4 Structure of CSCS

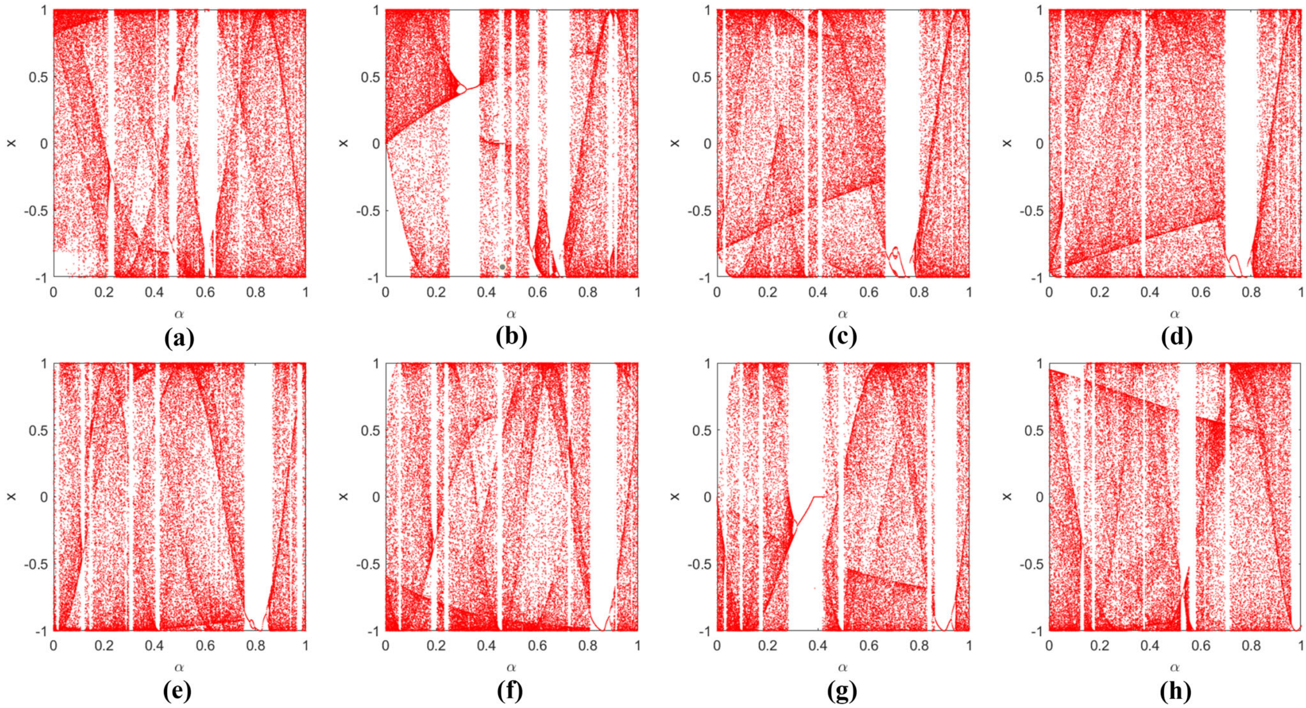
Aiming at the shortcomings of existing chaotic maps in weak dynamic behavior, a new 1-D compound Sine chaotic system (CSCS) is proposed. The CSCS is represented by the following equation.

$$x_{n+1} = \lvert\beta\pi \sin\{\pi[\mathcal{U}_1(e, x_n) + \mathcal{U}_2(f, x_n) + \theta]\}\rvert \quad (4)$$

where $\mathcal{U}_1(e, x_n)$ and $\mathcal{U}_2(f, x_n)$ are two common chaos maps in Sect. 2.1. The parameter θ is variable, and e, f and β are their control parameters. Considering the periodicity of Sine function, we set $\theta \in [-1, 1]$. In the following analysis, we set $e = \alpha$, $f = 1 - \alpha$, $\theta = 0.5$ and $\beta > 0$. Since the two maps in the CSCS may be any common 1-D chaos map, researchers can flexibly use two different existing maps to generate many novel chaos maps.

By the use of two of the four common maps in Sect. 2.1, we can get six 1-D novel chaos maps combining Eq. (4). And here we introduce four chaotic maps as shown in Table 2. For example, we can make use of the Cubic map and the Logistic map to produce the Logistic-Cubic-Sine (LCS) map. Similarly, we can generate the LSS map, the STS map and the TLS map.

Then, we begin to illustrate the excellent performance of the CSCS. Here, we explain it from the following two aspects: the bifurcation diagram and the Lyapunov exponent. By changing the control parameter β, we draw eight bifurcation diagrams and eight LE spectra of the LCS map generated by the CSCS. For the convenience of comparison, these bifurcation diagrams are normalized. In other words, the value range of x falls between 0 and 1. As shown in Figs. 3 and 4, when the control parameter β is small, for example, $\beta = 0.5$, the whole data range cannot be randomly covered by the output of the LCS map and the LEs of the LCS map are relatively small and not all greater than 0; the whole data range can be randomly covered by the output of



(a) **(b)** **(c)** **(d)**

(e) **(f)** **(g)** **(h)**

Fig. 1 Bifurcation diagrams of the LCC mapping: **a** $\gamma = -0.8$; **b** $\gamma = -0.5$; **c** $\gamma = -0.2$; **d** $\gamma = -0.1$; **e** $\gamma = 0.1$; **f** $\gamma = 0.3$; **g** $\gamma = 0.5$; break **h** $\gamma = 0.9$

Fig. 2 LE distributions of the LCC mapping

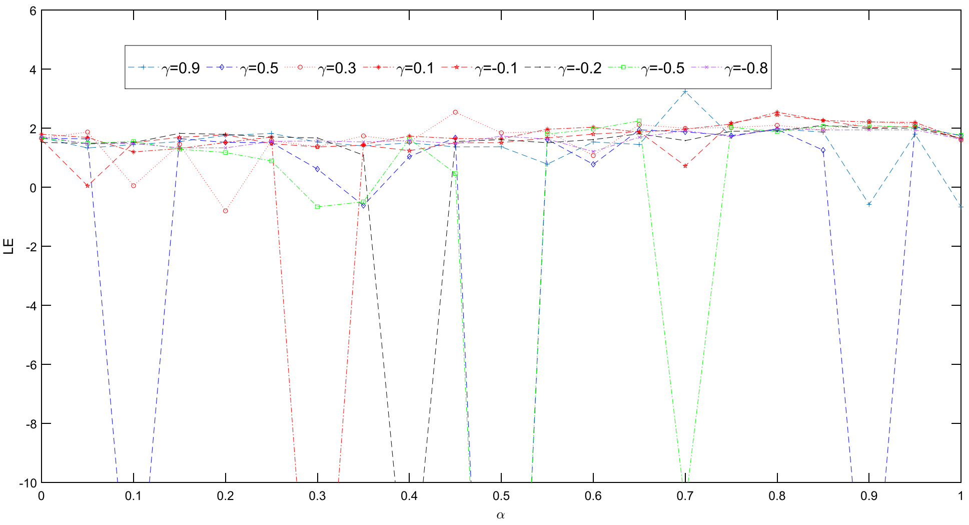


Table 2 Four new chaotic maps generated by the CSCS

$\mathcal{U}_1(e, x_n)$	$\mathcal{U}_2(f, x_n)$	Map definitions
$\mathcal{L}(\alpha, x_n)$	$\mathcal{S}(f^*, x_n)$	LSS: $x_{n+1} = \lvert\pi\beta\sin\{\pi[4\alpha x_n(1-x_n) + f\sin(\pi x_n) + 0.5]\}\rvert$
$\mathcal{S}(\alpha, x_n)$	$\mathcal{T}(f, x_n)$	STS: $x_{n+1} = \lvert\pi\beta\sin\{\pi[\alpha\sin(\pi x_n) + 1 - 2f\lvert x_n - 0.5f^{-1}\rvert + 0.5]\}\rvert$
$\mathcal{T}(\alpha, x_n)$	$\mathcal{L}(f, x_n)$	TLS: $x_{n+1} = \lvert\pi\beta\sin\{\pi[1 - 2\alpha\lvert x_n - 0.5\alpha^{-1}\rvert + 4f x_n(1-x_n) + 0.5]\}\rvert$
$\mathcal{L}(\alpha, x_n)$	$\mathcal{C}(f, x_n)$	LCS: $x_{n+1} = \lvert\pi\beta\sin\{\pi[4\alpha x_n(1-x_n) + \lvert 4x_n^3 - 3f x_n\rvert + 0.5]\}\rvert$

* $f = 1 - \alpha$

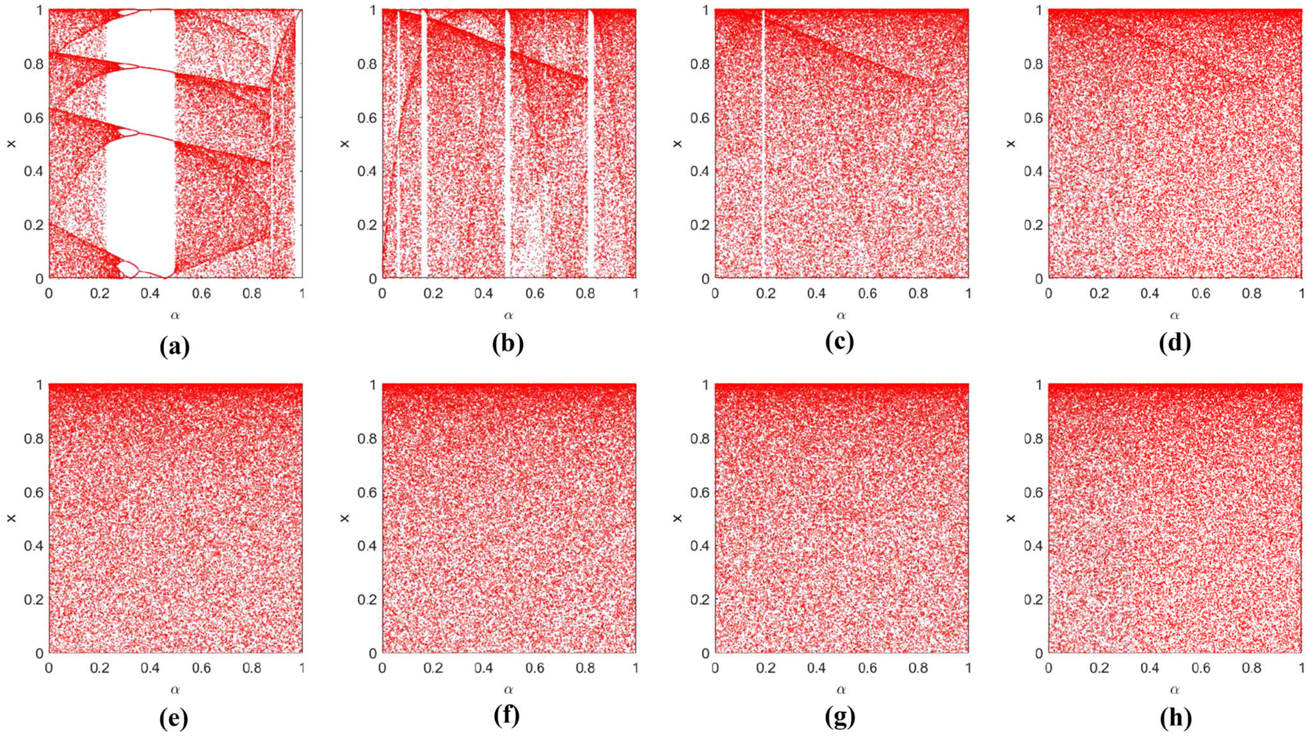


Fig. 3 Bifurcation diagrams of LCS: **a** $\beta = 0.1$; **b** $\beta = 0.3$; **c** $\beta = 0.5$; **d** $\beta = 1$; **e** $\beta = 5$; **f** $\beta = 10$; **g** $\beta = 20$; **h** $\beta = 50$

the LCS map and the LEs of the LCS map are getting bigger when we increase the control parameter β. When $\beta = 50$ and $\alpha \in [0, 1]$, the LEs of the LCS map are about 16.7. So we can increase the control parameter β to enhance the system performance. Shen *et al.* increased the dimension of chaotic system to obtain larger LE [38]. When the dimension of the system is 21, they can get a maximum LE of 15.3671. Obviously, our approach is simpler. Therefore, when a new chaotic map is generated by the CSCS using other existing 1-D maps as seed maps, we give an effective method to make the whole data range randomly covered by the output of the LCS map and to obtain larger LE.

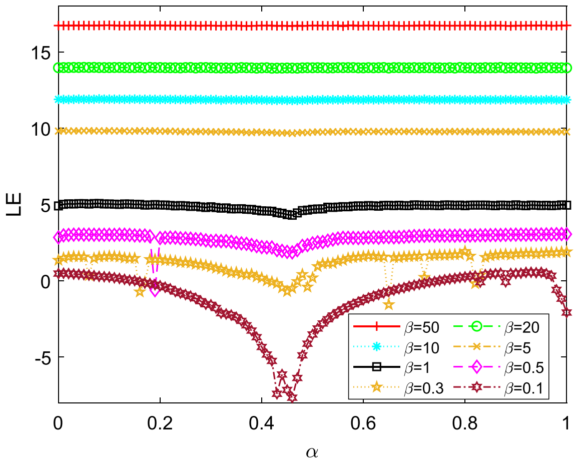


Fig. 4 LE distributions of LCS

2.5 Analyses on the system performance

To confirm the excellent performance about our chaos systems, we discuss performance comparison and analyses about four types of chaotic maps, namely, our method, Hua's method [13], Wang's method [32] and common maps in Sect. 2.1. Then, bifurcation diagrams, sample entropy [39], NIST SP800-22 [40] and LEs are used to evaluate its chaotic performance.

2.5.1 Bifurcation diagram

When the control parameters of a map change, the topology of the map will change accordingly, and a bifurcation may occur. Using the bifurcation diagram of a dynamic system, researchers can observe the chaotic behaviors of the dynamic system. These bifurcation diagrams are shown in Fig. 5 when α varies from 0 to 1. In the first row of Fig. 5, four bifurcation diagrams about LSS($\beta = 23$, Normalized), STS($\beta = 225$, Normalized), TLS($\beta = 8$, Normalized) and LCS($\beta = 1$, Normalized) are generated by the CSCS; Four bifurcation diagrams in the second row are generated by the CTBCS [13]; Four bifurcation diagrams in the third row are generated by discrete cascade chaos [32]. At the same time, we can see that only part of the phase planes is covered by the output states of

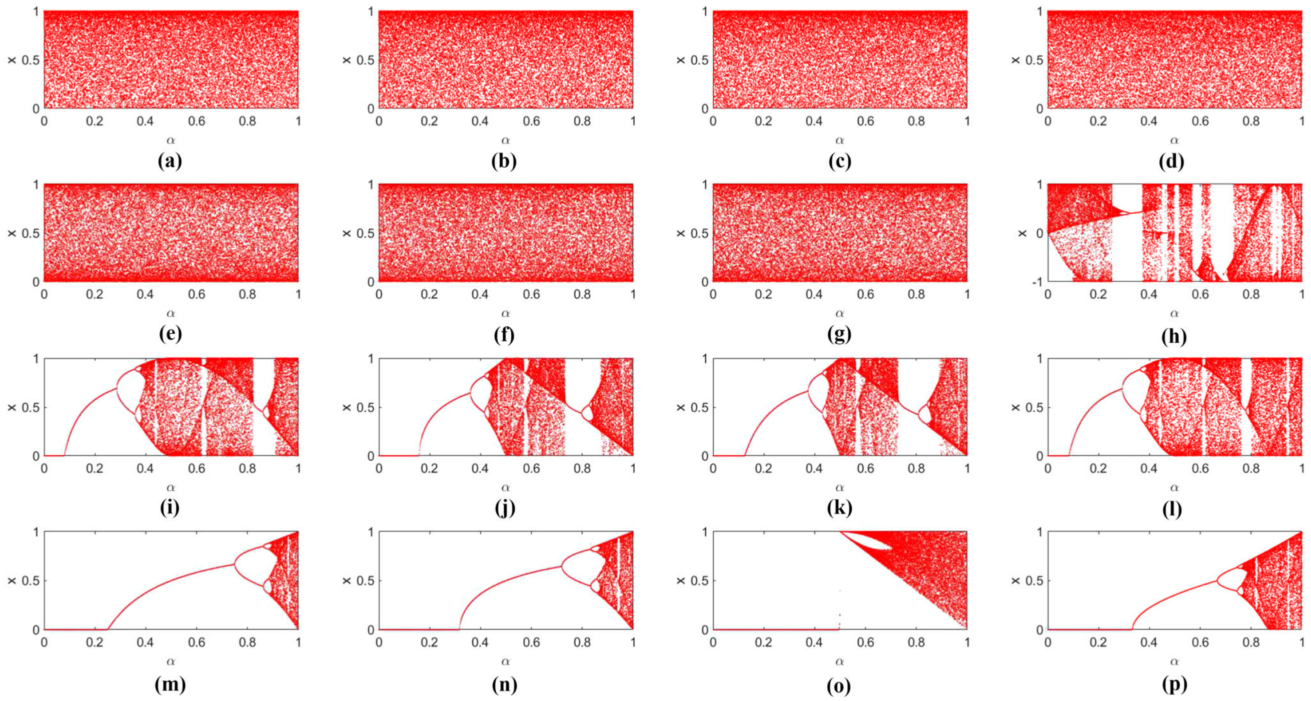


Fig. 5 Bifurcation diagrams of these maps: **a** LSS; **b** STS; **c** TLS; **d** LCS; **e** LSC; **f** STC; **g** TLC; **h** LCC; **i** LS; **j** ST; **k** LT; **l** LC; **m** Logistic; **n** Sine; **o** Tent; **p** Cubic

the four common maps in the fourth row of Fig. 5. Similarly, only part of the phase planes is covered by the output states about four cascade chaotic maps. Moreover, only part of the phase plane is covered by their output states of the LCC map generated by CTBCS, too. But the four maps generated by the CSCS are full mapping when α varies from 0 to 1. And the whole phase planes are randomly covered by the output states. In other words, this shows that the dynamic behaviors of these chaotic maps generated by our method are more robust and more random.

2.5.2 Lyapunov exponent

Chaotic dynamic systems have some important characteristics, such as its initial value sensitivity and unpredictability. In many chaotic evaluation techniques, the LE as a quantitative measurement method is used to judge whether a dynamic system is a chaotic system or not. We can use the following definition to illustrate the LE.

Definition 2 Given one-order difference equation $x_{n+1} = Q(x_n)$, if $Q(x)$ is differentiable, denoted as $Q'(x_n)$, then the LE is mathematically defined by

$$\lambda_{Q(x_n)} = \lim_{i \to \infty} \left[\frac{1}{i} \sum_{n=0}^{i-1} |Q'(x_n)| \right] \tag{5}$$

When the LE of a dynamical system is greater than 0, the dynamical system has chaotic behavior. And then the system exhibits non-chaotic behavior if it is negative or zero. The larger the LE is, the faster the trajectories diverge. And the corresponding chaotic system has better system performance. As shown in Fig. 6, we plot the LEs about 16 chaotic maps. By the use of the CSCS, we can get four chaos maps, namely the LSS map, the STS map, the TLS map and the LCS map. Based on the CTBCS, we can get three maps, namely the LSC map, the STC map and the TLS map [13]. When α is between 0 and 1, all LEs of the above seven maps are greater than 0. The LEs of the remaining 9 of the 16 maps are different from that of the above 7 maps. In other words, the LEs of these nine chaotic maps are sometimes greater than 0 and sometimes less than 0. In these four types of chaotic maps, that is, the maps generated by the CSCS, the maps generated by the CTBCS, the maps generated by cascade chaos and the common maps in Sect. 2.1, the first two have larger chaotic intervals than the last two. And we can observe that the maps generated by the CSCS have larger LEs compared with the maps generated by the CTBCS. Moreover, based on the analysis in Sect. 2.4, by increasing the control parameter β, the maps generated by the CSCS can get larger LEs. Therefore, these chaotic maps generated by our proposed CSCS have more random chaotic behaviors.

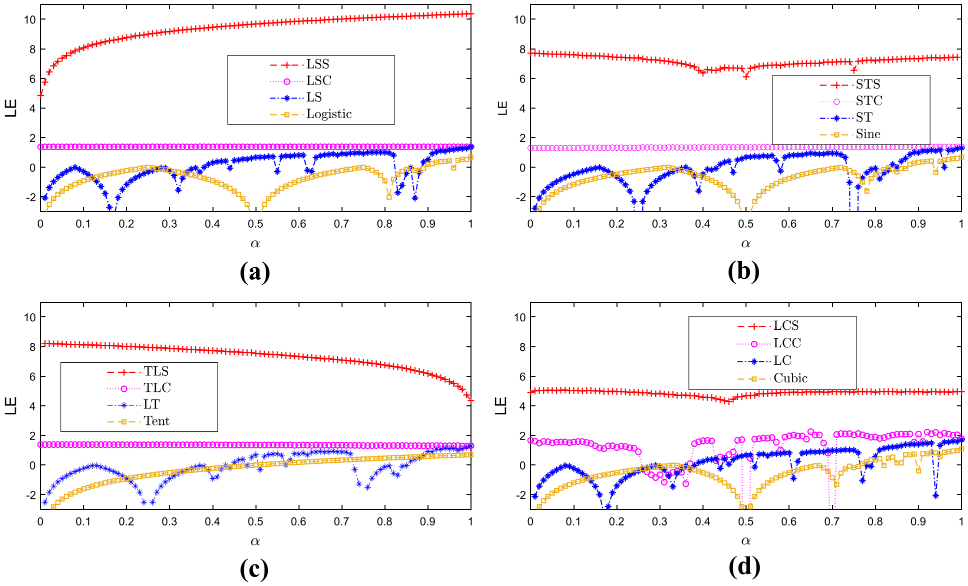


Fig. 6 LE comparisons of different chaos maps: **a** LSS, LSC , LS and Logistic; **b** STS, STC, ST and Sine; **c** TLS, TLC, LT and Tent; **d** LCS, LCC, LC and Cubic

2.5.3 Sample entropy

At present, there are many methods to survey the regularity of a chaotic series, for example, the Kolmogorov entropy(KE) [41,42], the approximate entropy(AE) [43,44] and the sample entropy (SE)[39]. According to these research work [41,45], a novel theory called AE was put forward to survey the regularity about a chaotic series. However, the AE has two obvious shortcomings, that is, the heavy reliance on the length of chaotic sequence and the lack of relative consistency [39]. In order to make up for these deficiencies, a new method called SE was put forward.

The SE effectively measures two aspects of discrete time series, namely complexity and randomness. When the SE of a dynamical system is greater than 0, then the dynamical system is a chaotic system. The larger the SE is, the more random the chaos series is. We can use the following definition to illustrate the SE.

Definition 3 For a u-dimensional time series $\mathbf{P} = \{p_1, p_2, \ldots, p_W\}$, the SE is given as follow:

$$SE(u, v, W) = -\log \frac{E_1}{E_2} \tag{6}$$

where the template vector $\mathbf{P}_u(i) = \{p_i, p_{i+1}, \ldots, p_{i+u-1}\}$. And $G[\mathbf{P}_u(i), \mathbf{P}_u(j)]$ denotes the distance between the template vector $\mathbf{P}_u(i)$ and the template vector $\mathbf{P}_u(j)$ [44]. Let E_1 be the amount of vectors with $G[\mathbf{P}_{u+1}(i), \mathbf{P}_{u+1}(j)] < v$. Similarly, E_2 denotes the amount of vectors with $G[\mathbf{P}_u(i), \mathbf{P}_u(j)] < v$. Here we make u equal to 2 and v equal to 0.2 times of the standard deviation about the chaotic sequence.

In Fig. 7, we compare the SEs of 16 different chaotic maps. Obviously, in general, we can see that the four chaotic maps designed utilizing our proposed method possess the biggest SEs. This means that our proposed method can develop chaotic maps with more random chaotic sequences.

2.5.4 NIST statistical test

A chaos pseudo-random series must satisfy the random characteristic of cryptography system. The chaos pseudo-random series must be inestimable for image encryption system. In order to illustrate that it is suitable for applying the maps designed utilizing the CSCS to image encryption, we use NIST SP800-22 to test the randomness of time series of the four maps generated by the CSCS. There are 15 sub-tests about NIST SP800-22 to examine the randomness of chaos series. Then, we will calculate the value of parameter p about the four new sequences produced by the CSCS. When the value of parameter p is greater than 0.01, we think the test is passed [40]. According to NIST's recommendations, we test at least 100 binary streams. And the length of each binary stream is 1000000 bits. As shown in Table 3, the binary sequences generated using LSS($\alpha = 0.2$), STS($\alpha = 0.8$), TLS($\alpha = 0.8$), and LCS($\alpha = 0.8$) can pass 15 sub-tests. In other words, the maps generated by the CSCS can generate pseudo-random series, which can be applied to image encryption.

3 Chaotic image cryptosystem

When a chaotic mapping is applied to cryptography, the security level about the cryptography system is seriously affected

Fig. 7 SE comparisons of different chaos maps: **a** LSS, LSC, LS and Logistic; **b** STS, STC, ST and Sine; **c** TLS, TLC, LT and Tent; **d** LCS, LCC, LC and Cubic

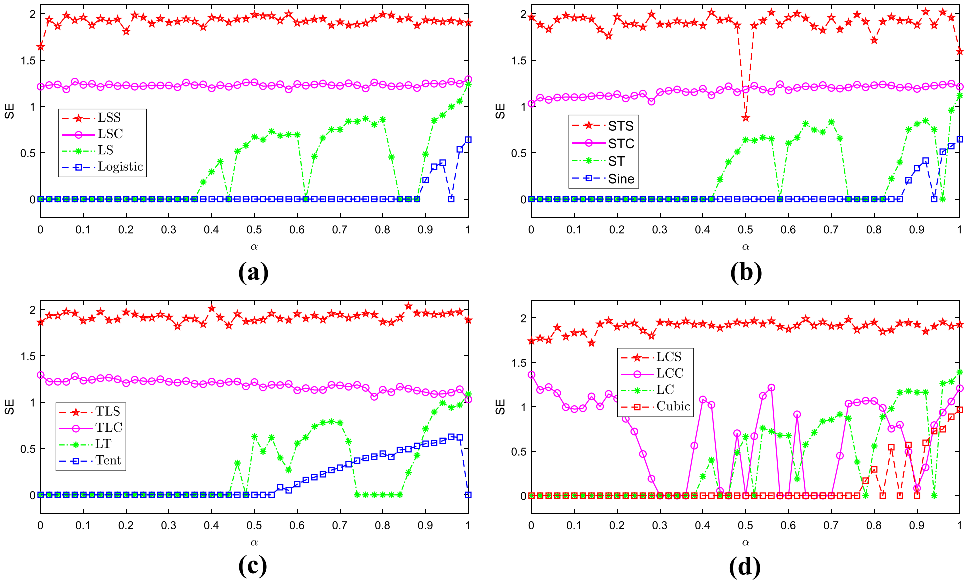


Table 3 NIST test results about four sequences produced using LSS($\alpha = 0.2$), STS($\alpha = 0.8$), TLS($\alpha = 0.8$), LCS($\alpha = 0.8$)

Sub-tests	LSS($p \geq 0.01$)	STS($p \geq 0.01$)	TLS($p \geq 0.01$)	LCS($p \geq 0.01$)
Frequency	0.066882(Pass)	0.978072(Pass)	0.071177(Pass)	0.883171(Pass)
BlockFrequency(M =128)	0.01265(Pass)	0.678686(Pass)	0.262249(Pass)	0.883171(Pass)
CumulativeSums Forward	0.010988(Pass)	0.474986(Pass)	0.971699(Pass)	0.249284(Pass)
CumulativeSums Reverse	0.12962(Pass)	0.032923(Pass)	0.964295(Pass)	0.678686(Pass)
Runs	0.304126(Pass)	0.699313(Pass)	0.55442(Pass)	0.534146(Pass)
LongestRun	0.035174(Pass)	0.437274(Pass)	0.55442(Pass)	0.719747(Pass)
Rank	0.191687(Pass)	0.574903(Pass)	0.319084(Pass)	0.678686(Pass)
FFT	0.759756(Pass)	0.574903(Pass)	0.249284(Pass)	0.867692(Pass)
NonOverlappingTemplatee (m= 9)*	0.308136(Pass)	0.435163(Pass)	0.493097(Pass)	0.511969(Pass)
OverlappingTemplatee(m=9)	0.171867(Pass)	0.202268(Pass)	0.019188(Pass)	0.362706(Pass)
Universal	0.145326(Pass)	0.455937(Pass)	0.911413(Pass)	0.978072(Pass)
ApproximateEntropy(m=10)	0.739918(Pass)	0.319084(Pass)	0.153763(Pass)	0.401199(Pass)
RandomExcursions *	0.380436(Pass)	0.456821(Pass)	0.502745(Pass)	0.494372(Pass)
RandomExcursionsVariant *	0.37455(Pass)	0.469017(Pass)	0.538446(Pass)	0.337957(Pass)
Seriale(m=16) p-value1	0.224821(Pass)	0.494392(Pass)	0.108791(Pass)	0.494392(Pass)
Seriale(m=16) p-value2	0.911413(Pass)	0.383827(Pass)	0.719747(Pass)	0.657933(Pass)
LinearComplexity(M =500)	0.419021(Pass)	0.262249(Pass)	0.798139(Pass)	0.616305(Pass)

* The average values of multiple tests

by the complexity of the chaos map. In this section, we use the LSS map and the LCS map in Sect. 2.4 to develop a new image encryption scheme, namely LSSLCS-based image encryption scheme (LSSLCS-IES). As we can see from Fig. 8, we give the flow chart of our new image encryption scheme. In order to make our encryption algorithm have excellent encryption effect, we design four scrambling operations, two image rotations and two diffusions in the encryption scheme. To facilitate the interpretation about our encryption scheme, the definitions of the LSS map and the LCS map are rewritten as follows:

$$x_{n+1} = |\pi\beta_1 \sin\{\pi[4\alpha_1 x_n(1-x_n) + (1-\alpha_1)\sin(\pi x_n) + \theta_1]\}| \quad (7)$$

$$x_{n+1} = |\pi\beta_2 \sin\{\pi[4\alpha_2 x_n(1-x_n) + |4x_n^3 - 3(1-\alpha_2)x_n| + \theta_2]\}| \quad (8)$$

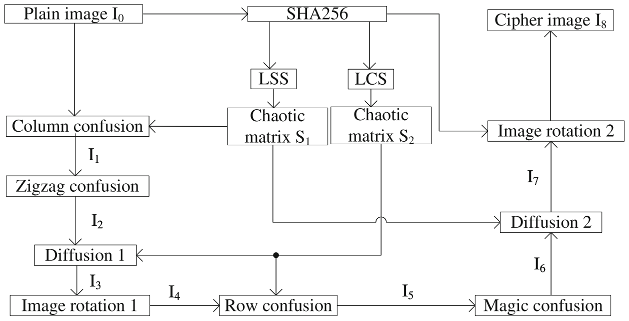


Fig. 8 The flow chart of the proposed LSSLCS-IES

3.1 Key generation

3.1.1 Secret key structure

As shown in Fig. 9, there are ten parts about the secret key of cryptosystem, including six system control parameters $\alpha_1 \in [0, 1]$, $\beta_1 > 0$, $\theta_1 \in [-1, 1]$, $\alpha_2 \in [0, 1]$, $\beta_2 > 0$, $\theta_2 \in [-1, 1]$, two system initial values $x_0 \in (0, 1]$, $y_0 \in (0, 1]$, parameters N_d and N_r . Here N_d represents the number of discarded elements. And N_r denotes the rotating parameter in Image rotation 2 of the block diagram of LSSLCS-IES.

3.1.2 Generation of secret key

In order to expand the key space and enhance the ability to resist the chosen plaintext attack, we utilize SHA-256 function to generate the key. First of all, using SHA-256 function based on an original image, we can obtain 256-bit hash value **K**. Then, we divide **K** into 32 blocks, each of which is 8 bits. The detailed description is as follows:

$$\mathbf{K} = \mathbf{k}_1\mathbf{k}_2 \ldots \mathbf{k}_{32}, \mathbf{k}_i = k_{i1}k_{i2} \ldots k_{i8} \tag{9}$$

where $1 \leq i \leq 32$. Then six system control parameters, namely, α_1, β_1, θ_1, α_2, β_2, and θ_2, are updated as follows:

$$\alpha_1 = \alpha_1 + 0.001 \times ((\mathbf{k}_1 \oplus \mathbf{k}_2) \oplus \mathbf{k}_3) \div 255 \tag{10}$$

$$\beta_1 = \beta_1 + 0.001 \times ((\mathbf{k}_4 \oplus \mathbf{k}_5) \oplus \mathbf{k}_6) \div 255 \tag{11}$$

$$\theta_1 = \theta_1 + 0.001 \times ((\mathbf{k}_7 \oplus \mathbf{k}_8) \oplus \mathbf{k}_9) \div 255 \tag{12}$$

$$\alpha_2 = \alpha_2 + 0.001 \times ((\mathbf{k}_{10} \oplus \mathbf{k}_{11}) \oplus \mathbf{k}_{12}) \div 255 \tag{13}$$

$$\beta_2 = \beta_2 + 0.001 \times ((\mathbf{k}_{13} \oplus \mathbf{k}_{14}) \oplus \mathbf{k}_{15}) \div 255 \tag{14}$$

$$\theta_2 = \theta_2 + 0.001 \times ((\mathbf{k}_{16} \oplus \mathbf{k}_{17}) \oplus \mathbf{k}_{18}) \div 255 \tag{15}$$

where the symbol $\oplus$ denotes the bitxor operation. And we use Eqs. (16) and (17) to update initial values, namely x_0 and y_0.

Table 4 Column confusion

```
Algorithm 1 Column confusion $\mathbf{I}_1 = CC(\mathbf{I}_0, \mathbf{S}_1)$
Input: Plain image matrix $\mathbf{I}_0$ and normalized chaotic matrix $\mathbf{S}_1$ with size of $M \times N$.
Output: Column confusion result $\mathbf{I}_1$.
1. Sort each column of $\mathbf{S}_1$ with ascending order and obtain the index matrix $\mathbf{O}_1$;
2. for $j = 1$ to $N$ do
3.    for $i = 1$ to $M$ do
4.       $\mathbf{I}_1(i, j) = \mathbf{I}_0(\mathbf{O}_1(i, j), j)$ ;
5.    end for
6. end for
```

$$x_0 = x_0 + 0.001 \times ((\mathbf{k}_{19} \oplus \mathbf{k}_{20}) \oplus \mathbf{k}_{21}) \div 255 \tag{16}$$

$$y_0 = y_0 + 0.001 \times ((\mathbf{k}_{22} \oplus \mathbf{k}_{23}) \oplus \mathbf{k}_{24}) \div 255 \tag{17}$$

In our encryption scheme, we also use two other parameters, namely N_d and N_r, as the security key. Here the size of the original gray image is $M \times N$ and the image matrix is denoted as $(\mathbf{I}_0)_{M\times N}$. Using Eqs. (7) and (8), we iterate the two normalized chaotic systems $(M \times N + N_d)$ times. When the former N_d elements of original sequences are discarded, we can generate two novel sequences with $M \times N$ elements. Then, the two new sequences are converted into the matrix $(\mathbf{S}_1)_{M\times N}$ and the matrix $(\mathbf{S}_2)_{M\times N}$. These two parameters are defined as follows:

$$N_r = (\mathbf{k}_{25} \oplus \mathbf{k}_{26}) \times M \div 4 \tag{18}$$

$$N_d = \mathbf{k}_{27} + \mathbf{k}_{28} + \mathbf{k}_{29} + \mathbf{k}_{30} + \mathbf{k}_{31} + \mathbf{k}_{32} \tag{19}$$

3.2 Image encryption scheme

Then, we will introduce the detailed process of our image encryption scheme as shown in Fig. 8.

3.2.1 Column confusion

According to the normalized chaos matrix $\mathbf{S}_1$ produced by the LSS map, these pixel positions about a plain image are randomly shuffled by use of the column confusion. The column confusion is defined by

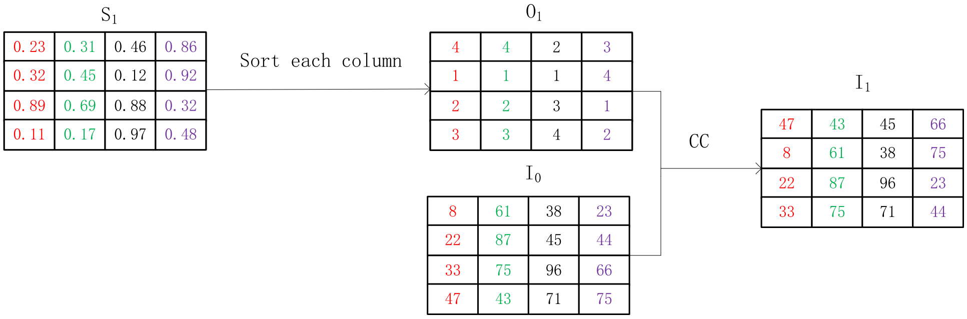
$$\mathbf{I}_1 = CC(\mathbf{I}_0, \mathbf{S}_1) \tag{20}$$

And then we use Table 4 for explaining the detailed process of the column confusion operation. At the same time, we give a numerical example about the column confusion as shown in Fig. 10. Here $\mathbf{O}_1$ is the index matrix associated with the chaotic matrix $\mathbf{S}_1$. And the pixels are shuffled through the column confusion operation.

Fig. 9 Secret key structure

α_1
β_1
θ_1
α_2
β_2
θ_2
x_0
0
r

Fig. 10 A numerical example of column confusion



S1

0.23	0.31	0.46	0.86
0.32	0.45	0.12	0.92
0.89	0.69	0.88	0.32
0.11	0.17	0.97	0.48

O1

4	4	2	3
1	1	1	4
2	2	3	1
3	3	4	2

I0

8	61	38	23
22	87	45	44
33	75	96	66
47	43	71	75

I1

47	43	45	66
8	61	38	75
22	87	96	23
33	75	71	44

Table 5 Zigzag confusion

```
Algorithm 2 Zigzag confusion $\mathbf{I}_2 = ZC(\mathbf{S}_1)$
Input: Image matrix $\mathbf{I}_1$ and the index matrix $\mathbf{O}_2$ with size of $M \times N$.
Output: Zigzag confusion result $\mathbf{I}_2$.
1. Using Zigzag transformation scan, obtain one-dimensional array $\mathbf{I}_{1V}$ and corresponding one-dimensional index array $\mathbf{O}_{2V}$ ;
2. for $i = 1$ to $M \times N$ do
3.     $\mathbf{I}_{1V2} = \mathbf{I}_{1V}(\mathbf{O}_{2V}(i))$ ;
4. end for
```

3.2.2 Zigzag confusion

Recently, some scrambling algorithms have been proposed based on Zigzag transform [46,47]. In the section, we propose a novel scrambling algorithm according to Zigzag transform. Zigzag confusion is defined by

$$\mathbf{I}_2 = ZC(\mathbf{I}_1) \tag{21}$$

The detailed process about Zigzag confusion is described in Table 5. And we give a numerical example in Fig. 11. In the j-th column and the i-th row in the image, we use $\mathbf{I}_1(i, j)$ to record the value of the pixel. And the calculation method of the corresponding index of this element is $(i-1) \times M + j$, denoted as $\mathbf{O}_2(i, j)$.

3.2.3 Diffusion 1

Firstly, we can get the diffusion matrix $\mathbf{S}_{2N}$ by the following equation.

$$\mathbf{S}_{2N} = mod(floor(\mathbf{S}_2 \times 10^{14}, 256) \tag{22}$$

Here the function floor is used to take the nearest integer value in the direction of negative infinity. And this is the result of mod operation to ensure that each element of $\mathbf{S}_{2N}$ is an integer and is in the range of [0, 255]. Then, we can obtain the image pixel matrix $\mathbf{I}_3$ by the following diffusion equation.

$$\mathbf{I}_3 = \mathbf{S}_{2N} \oplus \mathbf{I}_2 \tag{23}$$

3.2.4 Image rotation 1

In the section, by use of the rot90 function of MATLAB, the image $\mathbf{I}_3$ is rotated 90 degrees counterclockwise to get a new image pixel matrix $\mathbf{I}_4$.

3.2.5 Row confusion

According to the normalized chaos matrix $\mathbf{S}_2$ developed by the LCS map, these pixel positions about the image pixel matrix $\mathbf{I}_4$ are randomly shuffled by use of the row confusion. Here we use Eq. (24) to define the row confusion.

$$\mathbf{I}_5 = RC(\mathbf{I}_4, \mathbf{S}_2) \tag{24}$$

And then we use Table 6 for explaining the detailed process of the row confusion operation. At the same time, we give a numerical example about the row confusion as shown in Fig. 12. Here $\mathbf{O}_3$ is the index matrix associated with the chaotic matrix $\mathbf{S}_2$. And the pixels are shuffled through the row confusion operation.

3.2.6 Magic confusion

In the section, using the magic function of MATLAB, let's do Magic confusion operation. First of all, supposing that the size about the image $\mathbf{I}_5$ is $M \times N$, we use the following definition to explain the parameter P_M.

$$P_M = Max\{M, N\} \tag{25}$$

Then, we can get a magic matrix $\mathbf{M}_M$ based on the magic function. The generation rules of the index matrix $\mathbf{O}_4$ are as follows: If M equals N, let $\mathbf{O}_4 = \mathbf{M}_M$; if M is not equal to N, we discard $P_M \times P_M - M \times N$ elements in matrix $\mathbf{M}_M$ because the number of these elements about the magic matrix is greater than $M \times N$. As shown in Fig. 13, we give a numerical example about the generation the index matrix $\mathbf{O}_4$ when we let $M = 4$ and $N = 3$. Here Magic confusion is defined by

$$\mathbf{I}_6 = MC(\mathbf{I}_5, \mathbf{O}_4) \tag{26}$$

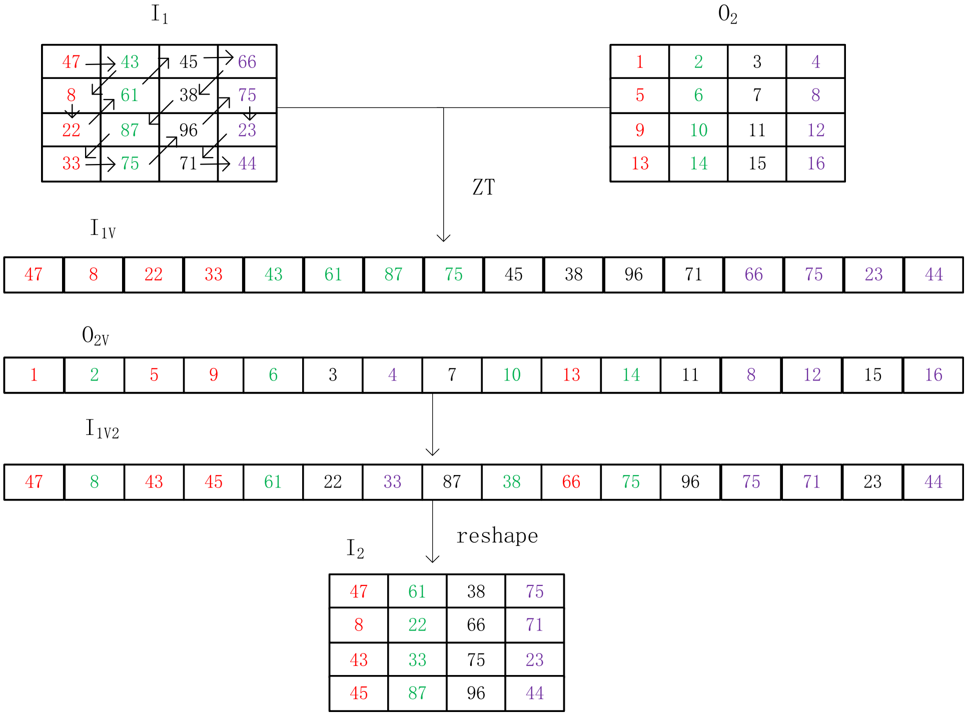


Fig. 11 A numerical example of Zigzag confusion

Table 6 Row confusion

```
Algorithm 3 Row confusion $\mathbf{I}_5 = RC(\mathbf{I}_4, \mathbf{S}_2)$
Input: Image matrix $\mathbf{I}_4$ and normalized chaotic matrix $\mathbf{S}_2$ with size of $M \times N$.
Output: Row confusion result $\mathbf{I}_5$.
Sort each row of $\mathbf{S}_2$ with ascending order and obtain the index matrix $\mathbf{O}_3$;
for $i = 1$ to $M$ do
    for $j = 1$ to $N$ do
        $\mathbf{I}_5(i, j) = \mathbf{I}_4(i, \mathbf{O}_3(i, j))$ ;
    end for
end for
```

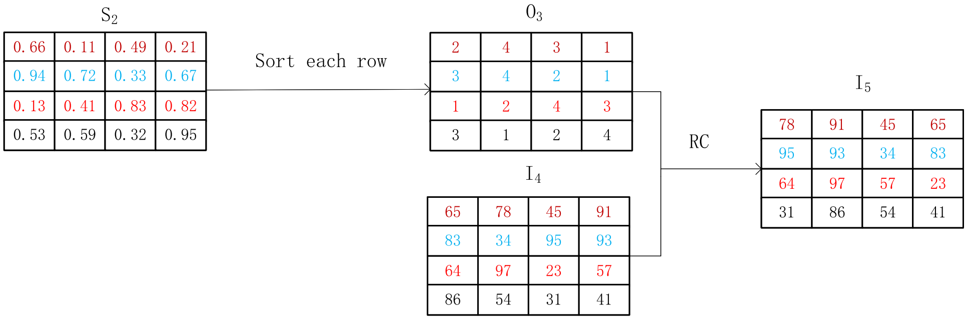


Fig. 12 A numerical example of row confusion

The detailed process of Magic confusion operation is described as shown in Table 7. And here we give a numerical example as shown in Fig. 14.

3.2.7 Diffusion 2

In the section, we can get the diffusion matrix $\mathbf{S}_{1N}$ by the following equation.

$$\mathbf{S}_{1N} = mod(floor(\mathbf{S}_1 \times 10^{14}, 256) \tag{27}$$

And we can get the cipher image pixel matrix $\mathbf{I}_7$ by the following diffusion equation.

$$\mathbf{I}_7 = \mathbf{S}_{1N} \oplus \mathbf{I}_6 \tag{28}$$

3.2.8 Image rotation 2

In the section, we firstly convert the gray scale image $\mathbf{I}_7$ into the 1-D image pixel array $\mathbf{S}_{I7}$

Fig. 13 A numerical example of the generation the index matrix $\mathbf{O}_4$

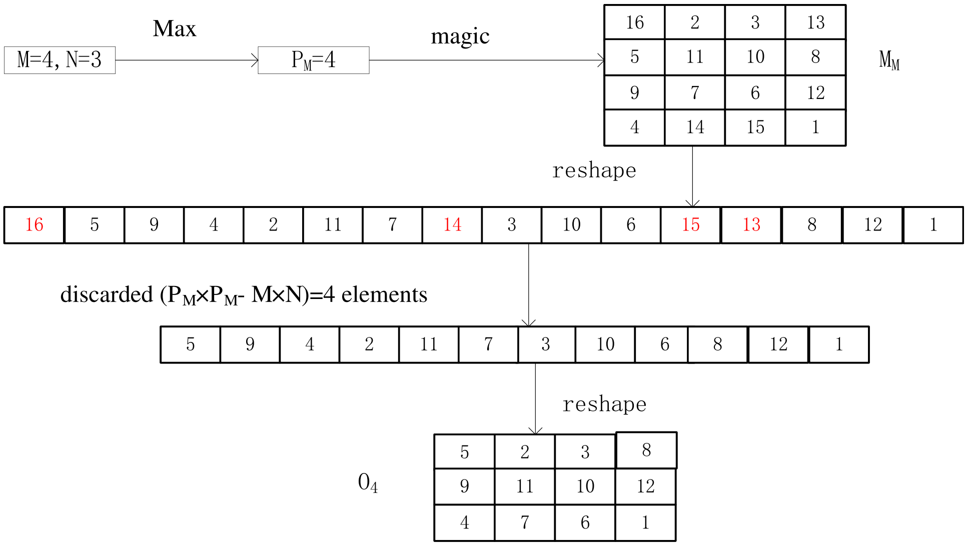


Table 7 Magic confusion

```
Algorithm 4 Magic confusion $\mathbf{I}_6 = MC(\mathbf{I}_5, \mathbf{O}_4)$
Input: Image matrix $\mathbf{I}_5$ and the index matrix $\mathbf{O}_4$ with size of $M \times N$.
Output: Magic confusion result $\mathbf{I}_6$.
1. $\mathbf{I}_{5V} = reshape(\mathbf{I}_5, 1, M \times N)$; $\mathbf{O}_{4V} = reshape(\mathbf{O}_4, 1, M \times N)$;
2. for $i = 1$ to $M \times N$ do
3.     $\mathbf{I}_{5V2} = \mathbf{I}_{5V}(\mathbf{O}_{4V}(i))$;
4. end for
5. $\mathbf{I}_6 = reshape(\mathbf{I}_{5V2}, M, N)$.
```

by the use of the reshape function of MATLAB. Then, we can obtain a new image pixel array $\mathbf{S}_{I8}$ by rotating the array $\mathbf{S}_{I7}$ to the left according to the number of N_r. And the security key N_r is defined by Eq. (18) in Sect. 3.1.2. $\mathbf{S}_{I8}$ can be calculated as follows:

$$\begin{cases} \mathbf{S}_{I8}(i - N_r) = \mathbf{S}_{I7}(i), & i - N_r \geq 1 \\ \mathbf{S}_{I8}(i - N_r + M \times N) = \mathbf{S}_{I7}(i), & else \end{cases} \tag{29}$$

The rotating operation can increase the strength of the image encryption [3]. Finally, we convert the image array $\mathbf{S}_{I8}$ into a new cipher image pixel matrix $\mathbf{I}_8$ by the use of the reshape function of MATLAB, too.

3.3 Image decryption scheme

We employ the same key and the same chaos series between the image encryption process and the image decryption process in our proposed scheme. At the same time, these confusion operations in the encrypted scheme are all reversible; the bitxor operation is also reversible. Based on these analyses, we know that its decryption process is the reverse process of image encryption. Considering the limitation of the length of our paper, we omit the detailed introduction of the decryption.

4 Experimental results and performance analysis

In the section, we have performed many experiments on general image sets to prove the effectiveness of our developed scheme. Here we choose some pictures from the USC-SIPI image database to do experiments [1]. MATLAB 2019b is utilized to implement the encryption and decryption program. The software and hardware configuration of the computer is as follows: $i7-7700HQ \quad CPU@2.80GHz$, $32\,GB$ memory and Windows 10 operating system. Compared to other advanced schemes, simulation tests show that our scheme not only has higher security but also faster encryption speed.

The specific experimental results are analyzed in detail as follows.

4.1 Simulation results

Three images with various characteristics and sizes are selected to illustrate our image encryption and decryption process. In the illustration experiment, we choose images 5.1.13(256 × 256), elaine.512(512 × 512), 7.2.01(1024 × 1024), All-black(512 × 512) and All-white(512 × 512) to show the universality of the proposed algorithm. We give the image encrypted results and the image decrypted results in Fig. 15. From Fig. 15, we can see that both the encrypted images and the original images are totally different. In other words, we cannot obtain any valid information about the plain images based on their encrypted images.

Generally speaking, digital images require a strong real-time property in the communication. So encryption efficiency is also one of the important performance indicators of an encryption scheme. In order to facilitate time complexity

[1] http://sipi.usc.edu/database/database.php.

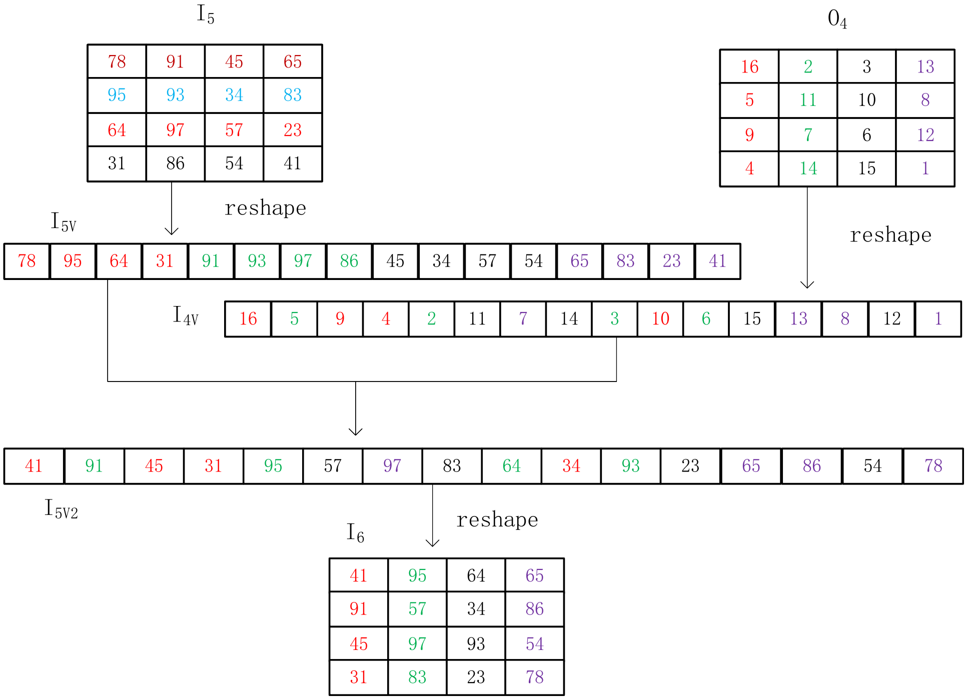


Fig. 14 A numerical example of Magic confusion

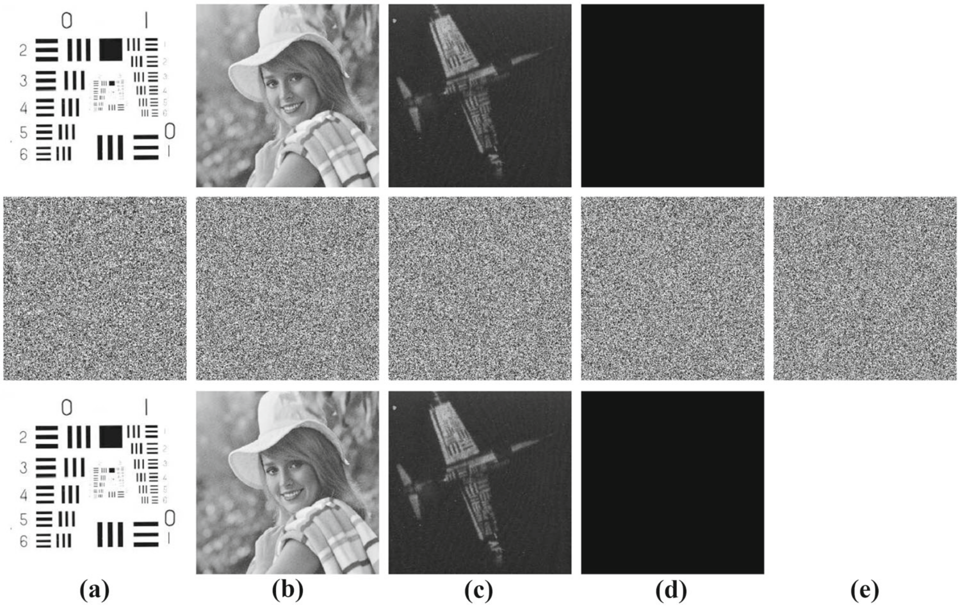


Fig. 15 Plain image, ciphered image, and decrypted image of the test images: **a** 5.1.13; **b** elaine.512; **c** 7.2.01; **d** All-black; **e** All-white

analysis, we assume that the encrypted image size is $M \times N$. The proposed encryption algorithm is mainly divided into two parts, namely the confusion operations and the diffusion operations. The time complexity analysis of the confusion mainly includes six parts: column confusion, Zigzag confusion, row confusion, magic confusion, rotation 1 and rotation 2. The total time complexity required for the confusion phase is calculated as $\Theta(M \times \log M + N \times \log N + 5 \times M \times N)$. The time complexity analysis of the diffusion mainly includes two parts: diffusion 1 and diffusion 2. The total time complexity of the diffusion phase is calculated as required $\Theta(2 \times M \times N)$. Therefore, the time complexity of the proposed encryption algorithm is $\Theta(M \times \log M + N \times \log N + 7 \times M \times N)$. The encryption time about our developed scheme is shown in Table 8, and comparisons with other algorithms are shown in Table 9. The given time for encrypting one image is the average value of 10 tests. As shown in Table 9, encryption time consumptions about our proposed scheme are the least, that is, encryption efficiency of our algorithm is the best.

Table 8 Encryption time consumptions (seconds) using our scheme

Test images	Size	Encryption time	Mean
5.1.09	256 × 256	0.06448	
5.1.10	256 × 256	0.06486	
5.1.11	256 × 256	0.06709	
5.1.12	256 × 256	0.06784	
5.1.13	256 × 256	0.06566	
5.1.14	256 × 256	0.06618	**0.06602**
5.2.08	512 × 512	0.23901	
5.2.09	512 × 512	0.25017	
5.2.10	512 × 512	0.24034	
7.1.01	512 × 512	0.23813	
7.1.02	512 × 512	0.23881	
7.1.03	512 × 512	0.24306	
7.1.04	512 × 512	0.24653	
7.1.05	512 × 512	0.24104	
7.1.06	512 × 512	0.23845	
7.1.07	512 × 512	0.24146	
7.1.08	512 × 512	0.24229	
7.1.09	512 × 512	0.24193	
7.1.10	512 × 512	0.24095	
boat.512	512 × 512	0.23363	
elaine.512	512 × 512	0.23366	
gray21.512	512 × 512	0.23795	
numbers.512	512 × 512	0.24081	
ruler.512	512 × 512	0.23768	**0.22289**
5.3.01	1024 × 1024	0.84873	
5.3.02	1024 × 1024	0.86111	
7.2.01	1024 × 1024	0.85177	
testpat.1k	1024 × 1024	0.85279	**0.85360**

Bold values indicate the average encryption time of pictures of the same size

Table 9 Encryption time consumptions (seconds) using different schemes

Image size	256 × 256	512 × 512	1024 × 1024
Ref. [48]	0.2224	0.9731	3.8377
Ref. [49]	0.1164	0.4924	2.0144
Ref. [50]	0.9810	3.8539	15.4564
Ref. [51]	0.3440	1.3357	5.3223
Ref. [13]	0.0949	0.4010	1.9857
Ref. [52]	0.1272	0.5156	2.1321
Ref. [14]	0.0779	0.3261	1.3146
Ours	0.0660	0.2229	0.8536

4.2 Security analysis

4.2.1 Key space

Having enough key space is one of the important performance indexes of a good encryption algorithm. The key space needs to be at least 2^{100} [25,53,54]. Theoretically, the larger key space provides stronger algorithm security. In our image encryption system, there are two types of keys as follows: (1) initial values x_0 and y_0, the control parameters α_1, β_1, θ_1, α_2, β_2, and θ_2; (2) the starting index N_d, the rotating index N_r.

If each of the first eight parameters has an accuracy of 10^{-14}, the first type of key space is $10^{14\times8}$. And we can get the starting index $N_d \in [0, 1530]$ and $N_r \in [0, 255 \times M \div 4]$ based on Sect. 3.1.2. M denotes the number of lines of an original image. The key space can be calculated as follows:

$$\begin{aligned} S_k &= 10^{14\times8} \times 1531 \times (255 \times M \div 4 + 1) \\ &\approx 1.5482M \times 2^{388} \gg 2^{100} \end{aligned} \tag{30}$$

Obviously, our proposed cryptosystem has enough key space to resist brute-force attack.

4.2.2 Statistical analyses

We discuss the resistance ability of an image encryption scheme for statistical attacks from the following four aspects: histogram, correlation, information entropy and local Shannon entropy [55]. In this part, we employ many experiments to prove the reliability of our designed algorithm as follows.

(1) Histogram

In digital images, the histogram represents the statistical relation of the number of occurrences about per gray level [56,57]. In order to resist statistical attacks, an encrypted image of an excellent encryption scheme must have a flat and uniform histogram. As we can see from Fig. 16, the three histograms of the plain images are unevenly distributed and fluctuates greatly, but the three corresponding encrypted images have almost flat histograms.

Next, we employ the histogram variance analysis and the chi-square test to evaluate uniformity of the image. Here, we assume that the encrypted image size is $M \times N$ and the range of image gray value is 0 to 255. Then, the variance of a histogram is defined as follows:

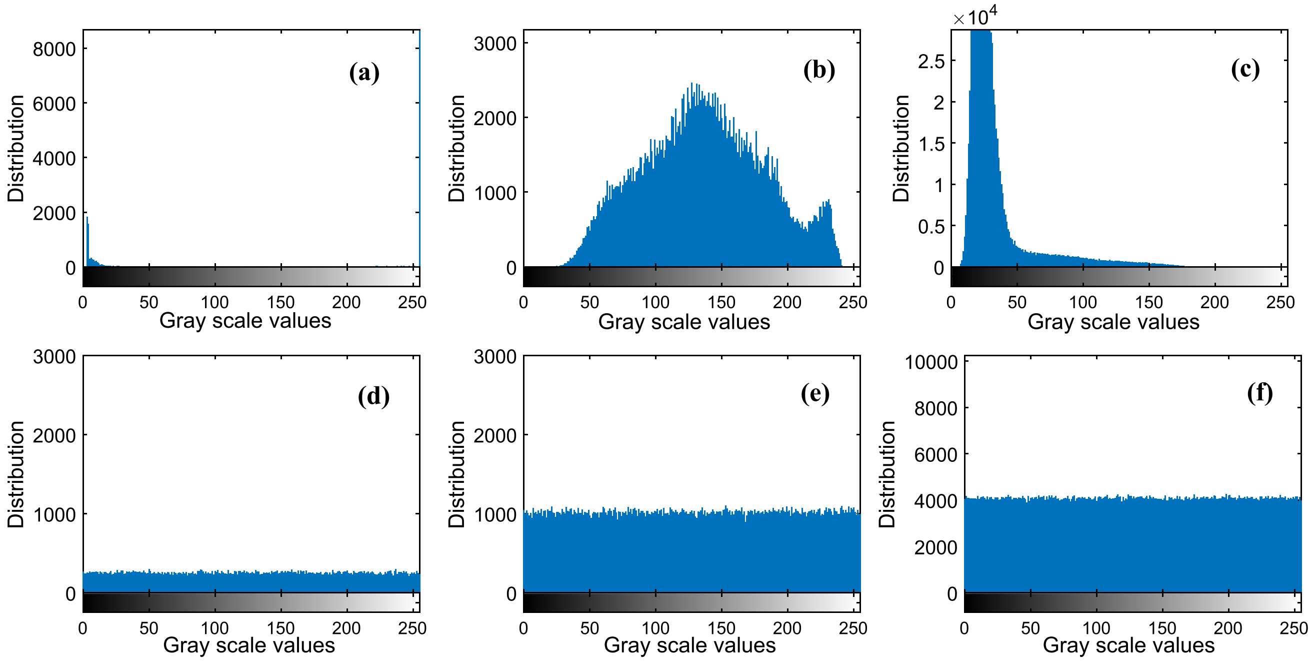


Fig. 16 Histograms: **a–c** 5.1.13, elaine.512, 7.2.01; **d–f** encrypted 5.1.13, encrypted elaine.512, encrypted 7.2.01

$$Var(Z) = \sum_{i=0}^{255}\sum_{j=0}^{255}\frac{(z_i - z_j)^2}{2\times 256^2} \tag{31}$$

and the Chi-square test (χ^2) is defined as follows:

$$\chi^2 = \sum_{k=0}^{255}\frac{(w_k - w_e)^2}{w_e} \tag{32}$$

where $Z = \{z_0, z_1, \ldots, z_{255}\}$ is the vector of the histogram values, $z_i(z_j)$ represents the number of pixels with gray value $i(j)$ in the image, w_k is the frequency of occurrence of pixel value k and $w_e = M \times N/256$.

When the value of the Var is lower, the grayscale uniformity of a histogram is better. That is, the effect of the cipher scheme is better [58]. When the test degree is 0.05, we can get the standard value $\chi^2_{0.05} = 293.2478$. If the calculated χ^2 result is less than the standard value, the Chi-square test is passed [17]. The calculated results about the Var and the Chi-square are listed in Table 10. On the one hand, the Var of the cipher image is obviously lower than that of the original image. On the other hand, the χ^2 of all encrypted images are smaller than the standard value, which means that the pixel value distributions of these encrypted images are uniform. Therefore, our encryption scheme has high security.

(2) Correlation between adjacent pixels

Generally speaking, having a correlation coefficient close to 0 is one of the important performance indicators of an excellent encryption scheme. Then, we give the three formulas to calculate the correlation coefficients from three directions, namely the horizontal direction, the vertical direction and the diagonal direction.

$$r_{xy} = \frac{cov(\eta, \rho)}{\sqrt{D(\eta)D(\rho)}} \tag{33}$$

$$cov(\eta, \rho) = \frac{1}{N}\sum_{i=0}^{N}(\eta_i - E(\eta))(\rho_i - E(\rho)) \tag{34}$$

$$D(\eta) = \frac{1}{N}\sum_{i=0}^{N}(\eta_i - E(\eta))^2, \quad E(\eta) = \frac{1}{N}\sum_{i=0}^{N}\eta_i \tag{35}$$

where we use η and ρ to record gray values about two adjoining pixels in a digital image. $E(\eta)$ represents its mean. $D(\eta)$ represents its variance. And $cov(\eta, \rho)$ represents the covariance of the corresponding pairs of pixel values.

As shown in Table 11, we can clearly see that the correlations are very small. There are about 40% of the cipher images whose absolute value of correlation coefficients is less than 0.0001. Moreover, the minimum correlation value of the ciphered test images is -2.45×10^{-5}. As we can see from Figs. 17, 18, 19, we draw the correlation coefficient graphs of three original images and three corresponding encrypted images. Obviously, it is very strong about the correlations of the three original images in three directions. But the points of these ciphered images are randomly covered in the entire space. Therefore, the effect of our encryption scheme is satisfactory.

Table 10 Var and χ^2 of the histogram

Test image	Size	Var		χ^2	
		Plain	Cipher	Plain	Cipher
5.1.13	256 × 256	11983209.89	248.37	11982953.89	248.37
elaine.512	512 × 512	562668.60	1075.39	110971.15	268.85
7.2.01	1024 × 1024	115198849.66	3734.17	6925496.10	233.39

Table 11 Correlation coefficients of encrypted images

Test images	Horizontal	Vertical	Diagonal
5.1.09	−0.0029	−0.0038	−0.0015
5.1.10	−0.0046	−5.56E−05	0.0032
5.1.11	0.0071	−0.0037	−0.0092
5.1.12	0.0041	0.0034	0.0018
5.1.13	0.0043	−0.0027	−0.000704
5.1.14	−0.0032	0.0025	−0.0039
5.2.08	0.00082	−0.000851	−0.0011
5.2.09	0.00055	0.0003749	0.0035
5.2.10	−0.0006	−0.0013	0.0025
5.3.01	0.0016	0.000567	−0.000482
5.3.02	0.00013	−0.000881	0.00038
7.1.01	−0.0003	0.0033	0.0011
7.1.02	−0.0012	0.0021	0.0012
7.1.03	0.0023	0.0021	−0.00067
7.1.04	0.00048	−0.000622	−0.000973
7.1.05	−2E−05	−0.000652	0.0006019
7.1.06	−0.0018	0.0009445	0.0002085
7.1.07	0.0019	0.0005669	0.0003404
7.1.08	0.0011	−0.0026	0.0017
7.1.09	0.0018	−0.0044	0.002
7.1.10	−0.0008	−0.0019	−0.000375
7.2.01	0.0025	−0.0012	0.0001143
boat.512	0.0006	0.0007502	−0.00047
elaine.512	0.0017	0.0037	−0.00028
gray21.512	0.0016	−0.000457	−0.0025
numbers.512	−0.0052	−5.93E−05	0.0017
ruler.512	−0.0012	0.0041	0.0021
testpat.1k	0.00023	−0.0023	0.0006386

(3) Information entropy

Generally speaking, we can use information entropy (IE) to describe the randomness of digital image information. For a whole image **S**, we can describe the mathematical definition of image IE utilizing the following equation.

$$H(\mathbf{S}) = -\sum_{i=0}^{2^n-1} P_1(s_i)\log_2(P_1(s_i)) \tag{36}$$

where $P_1(s_i)$ represents the probability about the occurrence of s_i. And s_i is the pixel value from 0 to 255 for an 8-bit gray image. In theory, a 256-level grayscale image has 2^8 possibilities of grayscale value. Therefore, the ideal value is equal to 8 about IE of this kind of image. In this sense, if the IE of the encrypted image obtained by a certain encryption scheme is closest to 8, then the effect of the encryption scheme is the best.

As we can see from Table 12, we give the IE results of plain images and the ciphered images utilizing three encryption algorithms. In the 28 test images, the scores of the IE of 20 encrypted images with our designed scheme are the largest. And the three average scores of the information entropy with our algorithm are also the largest for three different sizes of images.

In addition, we also made a comparative test with the other two image encryption algorithms. As shown in Table 13, the IE of the encrypted Lena obtained by our encryption algorithm is closest to 8. And we can see that the encrypted image is most random.

(4) Local Shannon entropy

Global Shannon entropy, that is the information entropy introduced in the previous chapter, has some known weaknesses, including the inaccuracy, the inconsistency and the low efficiency [55]. To overcome these weaknesses, the local Shannon entropy (LSE) has been proposed. The LSE is calculated as

$$H_{k,T_{\mathbf{B}}}(\mathbf{S}) = -\sum_{i=0}^{k} \frac{H(\mathbf{S}_{\mathbf{B}_i})}{k} \tag{37}$$

Here let's divide the whole image **S** into k sub image blocks. At the same time, any two sub image blocks cannot overlap. We use $\mathbf{B}_i (1 \leq i \leq k)$ to record sub-blocks. There are $T_{\mathbf{B}}$ pixels in each sub-block. In order to facilitate the comparative experiments with other encryption algorithms, we let $\alpha_3 = 0.05$, $k = 30$ and $T_{\mathbf{B}} = 1936$ [14]. When the score of the LSE about a ciphered image is between 7.901901305 and 7.903037329, the LSE test is passed [55]. According to the above settings, we can obtain its theoretical value, namely 7.902469317. As we can see from Table 14, we give the LSE results. From Table 14, we can see that the pass rate of our algorithm is 26/28, which is significantly higher than the

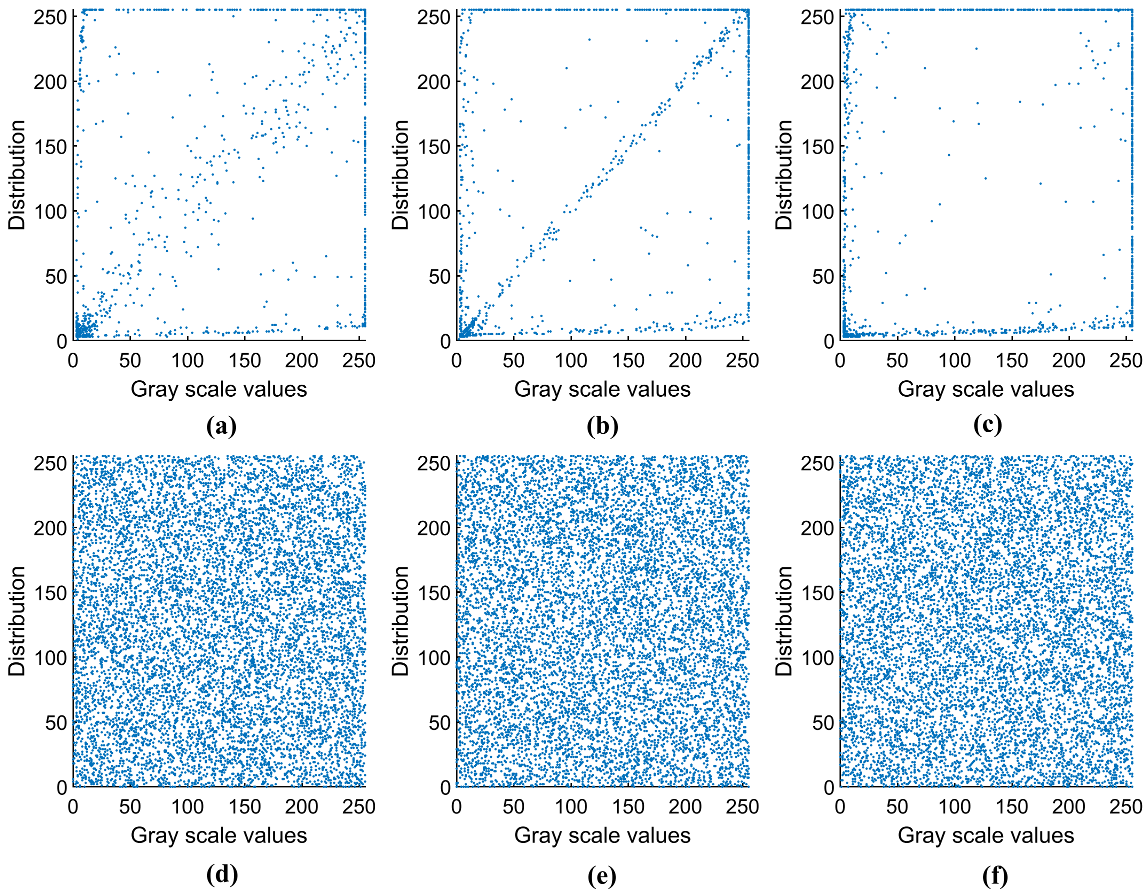


Fig. 17 Correlation of 5.1.13: **a**–**c** Horizontal, Vertical, Diagonal (5.1.13); **d**–**f** Horizontal, Vertical, Diagonal (ciphered 5.1.13)

values of 24/28 [14], 20/28 [52] and 17/25 [12]. At the same time, the average value of the LSE obtained by our algorithm is 7.902454 and the standard deviation is 0.000376. In the comparison results, the average value of the LSE about the cipher images produced by our proposed scheme is very close to 7.902469317. And the corresponding standard deviation is the smallest. Thus, our proposed scheme has higher security.

4.2.3 Differential attack analyses

In general, we discuss the resistance ability for differential attacks from the following two aspects: NPCR and UACI, namely the number of pixels change rate and the unified average changing intensity [60]. Suppose that C_1 and C_2 are different ciphered images, and there is only one different pixel value about their corresponding plain images. Here we can use the following two formulas to describe the definitions of NPCR and UACI.

$$NPCR = \sum_{i,j} \frac{D_d(i,j)}{M \times N} \times 100\% \tag{38}$$

$$UACI = \sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{M \times N \times \sigma} \times 100\% \tag{39}$$

where all images have the same size, that is $M \times N$. And σ represents the maximum pixel value. D_d records the difference between C_1 and C_2, which is defined as

$$D_d(i,j) = \begin{cases} 0, & \text{if } C_1(i,j) = C_2(i,j); \\ 1, & \text{if } C_1(i,j) \neq C_2(i,j). \end{cases} \tag{40}$$

We use I_{NPCR} to record the theoretical value of the NPCR, which is 99.6094 % [25]. Similarly, we use I_{UACI} to record the theoretical value of the UACI, which is 33.4635%. In addition, Wu *et al.* [61] developed more accurate criteria for the NPCR and the UACI. Based on a given significance level, namely α_4, we can use the following formula to calculate the critical value of the NPCR, namely $N^*_{\alpha_4}$.

$$N^*_{\alpha_4} = \frac{\sigma - \Phi^{-1}(\alpha_4)\sqrt{\sigma/(M \times N)}}{\sigma + 1} \tag{41}$$

where $\Phi(\alpha_4)$ represents its cumulative density function when the parameter α_4 obeys the standard normal distribution.

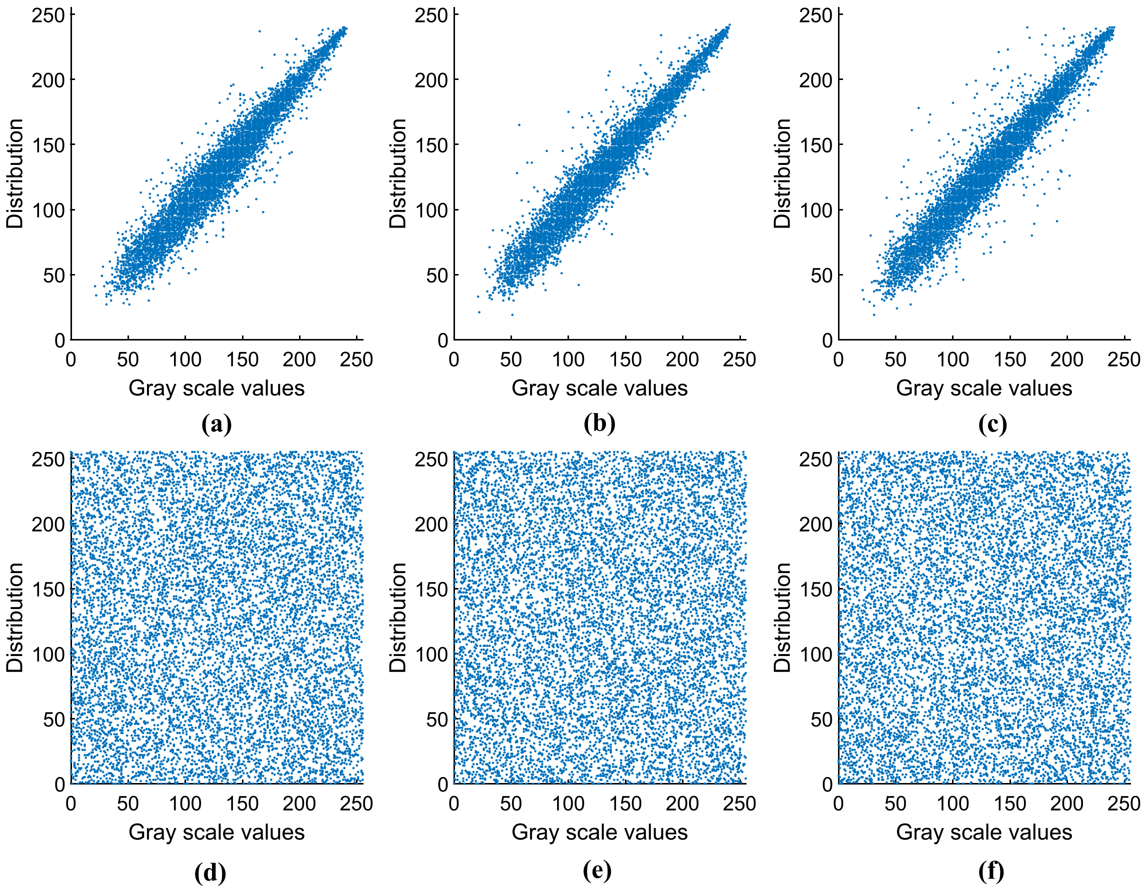


Fig. 18 Correlation of elaine.512: **a**–**c** Horizontal, Vertical, Diagonal (elaine.512); **d**–**f** Horizontal, Vertical, Diagonal (ciphered elaine.512)

And $\Phi^{-1}(\alpha_4)$ represents the corresponding inverse function. Then, we start to analyze the NPCR test. The NPCR test is considered to have passed if the NPCR score of an image encryption scheme is greater than $N^*_{\alpha_4}$. Similarly, the UACI test is considered to have passed if the UACI score of an image encryption scheme is between $U^{*-}_{\alpha_4}$ and $U^{*+}_{\alpha_4}$ [61].

In our experiment, we select the 28 gray scale images from the USC-SIPI image database. For the sake of generality, we randomly select a pixel in a plain image and then change its pixel value. And the test result of each image is the average of 100 test values. The specific rule is as follows:

$$s_i = \begin{cases} s_i + 1, & \text{if } s_i \neq 255; \\ s_i - 1, & \text{if } s_i = 255. \end{cases} \tag{42}$$

Here s_i is the pixel value. It is shown from the results about the NPCR test and the UACI test of our proposed scheme for different images in Table 15. There are an approximately 94.69 cent pass rate for the NPCR test and an approximately 92.11 cent pass rate for the UACI test when α_4 is equal to 0.05. In addition, by the use of different encryption scheme, the comparisons of the NPCR test and the UACI test of these encrypted images are shown as in Table 16. And our scheme can pass two types of tests for 28 images. At the same time, the average value of the NPCR about our scheme is the second closest to 99.6094%. Similarly, the theoretical value and the mean of the UACI about our scheme are the closest. Using our encryption scheme, we can get the minimum standard deviation of the NPCR and we can get the second smallest standard deviation of the UACI. In conclusion, the image encryption scheme designed by us has the strong resistance ability for differential attacks.

4.2.4 Key sensitivity

In general, a complex chaotic system can produce two completely different series when two initial conditions are slightly different. A good cipher scheme must be highly sensitive to the key. Initial value sensitivity is an important characteristic of chaotic mapping. Therefore, chaotic map is very suitable for encryption algorithm. The key of our algorithm involves 10 parameters.

To check the key sensitivity about our algorithm, the image called elaine.512 is firstly encrypted by use of the following key:

$K1$: x_0, y_0, α_1, β_1, θ_1, α_2, β_2, θ_2, N_d, N_r.

Then, we change one of the ten parameters with a tiny change and the rest remain unchanged during the encryption

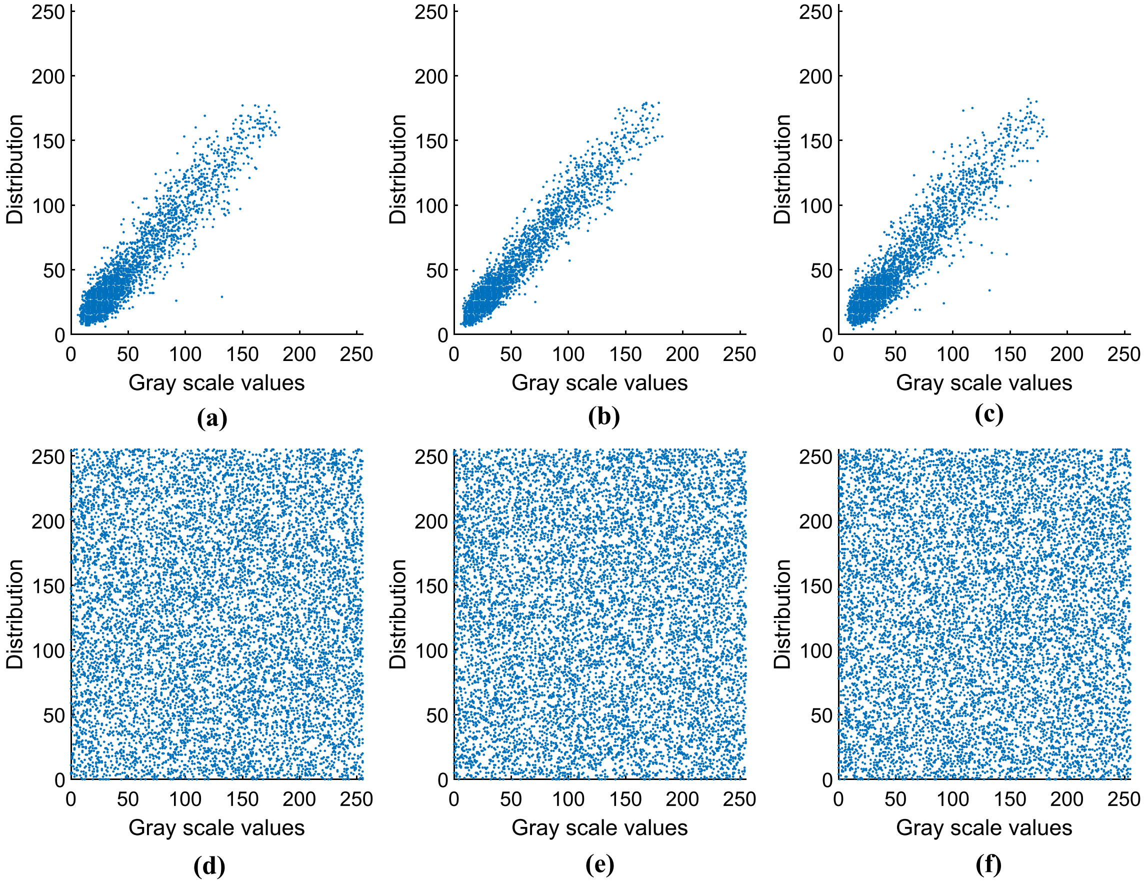


Fig. 19 Correlation of 7.2.01: **a**–**c** Horizontal, Vertical, Diagonal (7.2.01); **d**–**f** Horizontal, Vertical, Diagonal (ciphered 7.2.01)

process and the decryption process. Several modified keys are listed as follows:
$K2$: $x_0 + 10^{-14}$, y_0, α_1, β_1, θ_1, α_2, β_2, θ_2, N_d, N_r.
$K3$: x_0, $y_0 + 10^{-14}$, α_1, β_1, θ_1, α_2, β_2, θ_2, N_d, N_r.
$K4$: x_0, y_0, $\alpha_1 + 10^{-14}$, β_1, θ_1, α_2, β_2, θ_2, N_d, N_r.
$K5$: x_0, y_0, α_1, β_1, $\theta_1 + 10^{-14}$, α_2, β_2, θ_2, N_d, N_r.
$K6$: x_0, y_0, α_1, β_1, θ_1, α_2, $\beta_2 + 10^{-14}$, θ_2, N_d, N_r.

In the encryption process, by the use of six slightly different keys, we can get six encrypted images, that is, P_0, P_1, P_2, P_3, P_4 and P_5. As shown in Fig. 20, these six encrypted images are noise-like images. Here the NPCR and the UACI are used to quantitatively evaluate the key sensitivity about our proposed scheme [62,63]. According to Eqs. (38) and (39), we can obtain the values of the NPCR and the UACI between the encrypted image P_0 and the encrypted image P_1. Similarly, we can get the values of the NPCR and the UACI between the encrypted image P_0 and the encrypted image P_2 in Table 17. From Table 17, we can see that the values of the NPCR and the UACI are close to the ideal values, that is, $I_{NPCR} = 99.6094\%$ and $I_{UACI} = 33.4635\%$ [25]. This means that these encrypted images obtained with slightly different keys are almost completely different. In the decryption process, we decrypt the cipher image P_0 by the five keys, namely K2, K3, K4, K5, K6. As we can see from Fig. 21, the original image cannot be decrypted correctly as long as the key changes slightly. That is to say, our algorithm has high key sensitivity.

4.2.5 Randomness of encrypted images

To resist statistic attacks, the pixel distribution of an ideal encrypted image should obey uniform distribution. In Sect. 2.5.4, we used NIST SP800-22 to test the randomness of chaos series. Similarly, we employ NIST SP800-22 to test the randomness of encrypted images. The NIST test results are shown in Table 18. Due to the insufficient cycle number length of binary sequences of encrypted elaine.512 and encrypted 7.2.01, these two test items, namely “Random-Excursions” and “Random-Excursions Variant,” are not applicable [64]. At the same time, we can see that other test items have smoothly passed about these two encrypted images. To obtain binary sequences with sufficient number of cycles, we use the method in Ref. 13. In our test, 120 digital images, that is, filenames from 1.pgm to 120.pgm, are selected from the BOWS-2 database [2]. By the use of the LSSLCS-IES, we can get 120 encrypted images and

[2] http://bows2.ec-lille.fr/BOWS2OrigEp3.tgz.

Table 12 Information entropy analysis

Test image	Size	Original images	Encrypted images		
			Ref. [59]	Ref. [14]	Ours
5.1.09	256 × 256	6.7093	7.9966	7.9971	7.9974
5.1.10	256 × 256	7.3118	7.9971	7.9974	7.9974
5.1.11	256 × 256	6.4523	7.9975	7.9969	7.9977
5.1.12	256 × 256	6.7057	7.9972	7.9972	7.9971
5.1.13	256 × 256	1.5483	7.9965	7.9969	7.9973
5.1.14	256 × 256	7.3424	7.9977	7.9974	7.9972
Mean		6.01163	7.99710	7.99715	7.99735
5.2.08	512 × 512	7.2010	7.9991	7.9993	7.9994
5.2.09	512 × 512	6.9940	7.9992	7.9993	7.9991
5.2.10	512 × 512	5.7056	7.9991	7.9993	7.9992
7.1.01	512 × 512	6.0274	7.9990	7.9991	7.9993
7.1.02	512 × 512	4.0045	7.9991	7.9992	7.9993
7.1.03	512 × 512	5.4957	7.9990	7.9993	7.9993
7.1.04	512 × 512	6.1074	7.9992	7.9993	7.9994
7.1.05	512 × 512	6.5632	7.9992	7.9992	7.9993
7.1.06	512 × 512	6.6953	7.9992	7.9993	7.9993
7.1.07	512 × 512	5.9916	7.9991	7.9993	7.9992
7.1.08	512 × 512	5.0534	7.9990	7.9993	7.9993
7.1.09	512 × 512	6.1898	7.9991	7.9992	7.9993
7.1.10	512 × 512	5.9088	7.9990	7.9993	7.9993
boat.512	512 × 512	7.1914	7.9992	7.9994	7.9993
elaine.512	512 × 512	7.5060	7.9992	7.9993	7.9993
gray21.512	512 × 512	4.3923	7.9993	7.9994	7.9993
numbers.512	512 × 512	7.7292	7.9994	7.9991	7.9992
ruler.512	512 × 512	0.5000	7.9987	7.9992	7.9993
Mean		5.84759	7.99912	7.99927	7.99929
5.3.01	1024 × 1024	7.5237	7.9998	7.9998	7.9998
5.3.02	1024 × 1024	6.8303	7.9996	7.9998	7.9998
7.2.01	1024 × 1024	5.6415	7.9996	7.9998	7.9998
testpat.1k	1024 × 1024	4.4077	7.9998	7.9998	7.9998
Mean		6.10080	7.99970	7.99980	7.99980

Table 13 Comparison of information entropy

Test image	Size	Encrypted images		
		Ref. [6]	Ref. [9]	Ours
Lena.bmp	256 × 256	7.9970	7.9971	7.9972
Cameraman.bmp	256 × 256	7.9969	7.9967	7.9970

120 corresponding binary sequences. Then, the 120 binary sequences called ChenTZN are used as input in our test. At this time, the binary sequence is long enough [13]. As shown in Table 18, these encrypted images generated by our encryption scheme can pass all 15 sub-tests. In conclusion, the above analyses show that these encrypted image distributions are high random.

4.2.6 Four classical types of attacks

Generally speaking, there are four classical types of attacks, that is, ciphertext only, chosen plaintext, known plaintext and chosen ciphertext. We know that chosen plaintext attack is the most difficult attack to resist [65]. As long as an encryption system can resist the attack, it can resist the remaining three types of attacks. In order to enhance the ability to resist the chosen plaintext attack, we utilize SHA-256 function to generate the key. As shown in key sensitivity analysis, our cipher scheme is highly sensitive to initial values x_0, y_0, the control parameters α_1, β_1, θ_1, α_2, β_2, θ_2, the starting index N_d, and the rotating index N_r. Therefore, our proposed algorithm can resist the chosen plaintext attack.

Table 14 Comparative analysis of local Shannon entropy

Test image	Size	Ref. [12]	Ref. [52]	Ref. [14]	Ours
5.1.09	256 × 256	7.902388	*7.903369*	*7.903154*	7.902741
5.1.10	256 × 256	*7.900874*	*7.903520*	*7.901680*	7.902067
5.1.11	256 × 256	7.902073	7.902291	7.902725	7.902997
5.1.12	256 × 256	7.903020	7.902721	*7.901605*	7.902052
5.1.13	256 × 256	7.902777	7.902620	*7.901269*	*7.901183*
5.1.14	256 × 256	*7.903880*	7.902837	7.902341	7.902869
5.2.08	512 × 512	*7.904039*	7.902793	7.902038	7.902622
5.2.09	512 × 512	7.902479	7.902972	7.902722	7.902711
5.2.10	512 × 512	*7.900619*	7.902464	7.902478	7.902156
7.1.01	512 × 512	*7.903868*	7.902934	7.902012	7.902869
7.1.02	512 × 512	7.902903	7.902843	7.902484	7.902427
7.1.03	512 × 512	*7.904509*	*7.903339*	7.902833	7.902900
7.1.04	512 × 512	7.902012	7.902649	7.902047	7.902438
7.1.05	512 × 512	7.902417	7.902493	7.902568	7.902489
7.1.06	512 × 512	7.902155	*7.903261*	7.902022	7.902353
7.1.07	512 × 512	*7.899848*	7.902714	7.902398	7.902479
7.1.08	512 × 512	7.902132	7.902563	7.902137	*7.901770*
7.1.09	512 × 512	7.902149	*7.903185*	7.902142	7.902497
7.1.10	512 × 512	7.902141	7.902805	7.902171	7.902718
boat.512	512 × 512	7.902940	*7.903070*	7.902046	7.902477
elaine.512	512 × 512		7.902929	7.902632	7.902680
gray21.512	512 × 512	*7.903592*	*7.903238*	7.902718	7.902463
numbers.512	512 × 512		7.902697	7.902067	7.902761
ruler.512	512 × 512	7.902614	7.902755	7.902004	7.902466
5.3.01	1024 × 1024	7.902184	*7.903661*	7.902057	7.902479
5.3.02	1024 × 1024	7.902494	7.902545	7.902396	7.902460
7.2.01	1024 × 1024	7.902998	7.902896	7.902330	7.902421
testpat.1k	1024 × 1024		7.902715	7.902854	7.902160
Mean		7.902527	7.902889	7.902367	7.902454
STD		0.001058	0.0004	0.000408	0.000376
Pass/All		17/25	20/28	24/28	26/28

Italicized data indicates that the test failed

4.2.7 Encryption quality measurement

(1) Energy

The energy is the sum of the square of the element values of the gray level co-occurrence matrix (GLCM), which reveals the gray level distribution of the image. The lower the energy value, the more uniform the grayscale distribution of the image. It is defined as

$$Ene = \sum_{i,j} p^2(i, j) \tag{43}$$

where $p(i, j)$ is the number of GLCM matrices [66]. The energy calculation results of different encrypted images generated by different algorithms are shown in Table 19. From the results, the proposed encryption algorithm can effectively reduce the value of energy. At the same time, it has considerable encryption performance compared with other algorithms.

(2) Contrast

The contrast of an image reflects the clarity of the image, that is, the clarity of the texture. The greater the contrast, the deeper the texture grooves of the adjacent pixels of the image [67]. Therefore, a secure ciphertext image should have a large contrast to prove that its texture is non-homogeneous [68]. The contrast is defined as

$$Con = \sum_{i} \sum_{j} (i - j)^2 p(i, j) \tag{44}$$

where $p(i, j)$ is the number of GLCM matrices. The contrast calculation results are shown in Table 19. The results show

Table 15 Test analyses of NPCR (%) and UACI (%) for our scheme

Test image	NPCR				UACI			
	Max	Min	Pass(%)	Mean	Max	Min	Pass(%)	Mean
5.1.09	99.6567	99.5453	96	99.6118	33.6894	33.2053	94	33.4837
5.1.10	99.6719	99.5392	96	99.612	33.6521	33.1631	88	33.4216
5.1.11	99.6765	99.5575	97	99.6133	33.7307	33.2704	97	33.4748
5.1.12	99.6536	99.5209	96	99.6091	33.6609	33.2321	95	33.4644
5.1.13	99.6597	99.5575	90	99.6062	33.6348	33.2917	100	33.4585
5.1.14	99.6796	99.5514	96	99.6115	33.7633	33.3276	89	33.5271
5.2.08	99.6349	99.5762	98	99.6099	33.5584	33.3394	96	33.4627
5.2.09	99.641	99.5804	94	99.6098	33.5608	33.286	94	33.4418
5.2.10	99.6407	99.5712	94	99.6087	33.5644	33.3699	97	33.467
7.1.01	99.6445	99.5754	94	99.6095	33.5207	33.3087	80	33.4111
7.1.02	99.638	99.5808	95	99.6098	33.5699	33.3401	93	33.4598
7.1.03	99.6422	99.5728	93	99.6098	33.5679	33.3036	86	33.4458
7.1.04	99.6403	99.5773	96	99.6094	33.584	33.3287	97	33.4669
7.1.05	99.6422	99.5682	95	99.6099	33.5473	33.3307	95	33.4487
7.1.06	99.6346	99.5785	93	99.6068	33.5761	33.3645	97	33.4575
7.1.07	99.6361	99.5842	94	99.6109	33.5435	33.3345	99	33.4598
7.1.08	99.6376	99.5804	95	99.6085	33.5581	33.3593	98	33.4695
7.1.09	99.6399	99.5731	96	99.6099	33.5245	33.3117	96	33.4441
7.1.10	99.6357	99.5735	94	99.6091	33.5403	33.352	94	33.4519
boat.512	99.6326	99.5651	92	99.6103	33.6086	33.3805	94	33.481
elaine.512	99.6403	99.5792	96	99.6108	33.6093	33.3648	93	33.4804
gray21.512	99.6395	99.5762	94	99.6096	33.5613	33.3783	96	33.4873
numbers.512	99.6441	99.5831	96	99.6092	33.5426	33.3166	92	33.4375
ruler.512	99.6357	99.5853	95	99.6083	33.5878	33.3142	95	33.4617
5.3.01	99.6246	99.5982	97	99.6101	33.5147	33.4146	94	33.4691
5.3.02	99.6224	99.5876	94	99.609	33.5112	33.4097	94	33.4586
7.2.01	99.623	99.5945	94	99.6097	33.513	33.4098	98	33.4635
testpat.1k	99.6267	99.5954	94	99.6098	33.5009	33.3942	98	33.4588

Table 16 Test analyses of NPCR (%) and UACI (%) for different schemes

Test image	NPCR, $I_{NPCR} = 99.6094\%$				UACI, $I_{UACI} = 33.4635\%$			
	Ref. [12]	Ref. [52]	Ref. [14])	Ours	Ref. [12]	Ref. [52]	Ref. [14])	Ours
5.1.09	99.6064	99.603	99.6093	99.6118	33.4456	33.552	33.4723	33.4837
5.1.10	99.6154	99.636	99.6095	99.6120	33.4946	33.453	33.4663	33.4216
5.1.11	99.6244	99.942	99.6133	99.6133	33.5541	33.586	33.4554	33.4748
5.1.12	99.5703	99.792	99.6123	99.6091	33.4302	33.453	33.4604	33.4644
5.1.13	99.6109	99.792	99.6050	99.6062	33.4438	33.520	33.4601	33.4585
5.1.14	99.6364	99.621	99.6110	99.6115	33.4655	33.440	33.4606	33.5271
5.2.08	*99.5870*	99.960	99.6070	99.6099	33.4008	*33.692*	33.4734	33.4627
5.2.09	99.6260	99.876	99.6106	99.6098	33.4804	33.548	33.4572	33.4418
5.2.10	99.6124	99.654	99.6096	99.6087	33.4563	33.454	33.4575	33.4670
7.1.01	99.5992	99.957	99.6095	99.6095	33.5037	*33.648*	33.4726	33.4111

Table 16 continued

Test image	NPCR, $I_{NPCR} = 99.6094\%$				UACI, $I_{UACI} = 33.4635\%$			
	Ref. [12]	Ref. [52]	Ref. [14])	Ours	Ref. [12]	Ref. [52]	Ref. [14])	Ours
7.1.02	99.6075	99.918	99.6117	99.6098	33.4237	33.465	33.4563	33.4598
7.1.03	99.6079	99.849	99.6123	99.6098	33.4291	*33.273*	33.4535	33.4458
7.1.04	99.5988	99.991	99.6114	99.6094	33.4739	*33.202*	33.4475	33.4669
7.1.05	99.6170	99.942	99.6099	99.6099	33.4362	*33.830*	33.4559	33.4487
7.1.06	99.6272	99.670	99.6064	99.6068	33.3954	*33.627*	33.4515	33.4575
7.1.07	99.5931	99.983	99.6068	99.6109	33.4073	*33.609*	33.4638	33.4598
7.1.08	99.6094	99.818	99.6097	99.6085	33.4332	33.375	33.4536	33.4695
7.1.09	99.6162	99.874	99.6112	99.6099	33.4177	33.530	33.4729	33.4441
7.1.10	99.6045	99.697	99.6096	99.6091	33.4344	33.438	33.4605	33.4519
boat.512	99.6154	99.715	99.6084	99.6103	33.4654	33.374	33.4434	33.4810
elaine.512	99.6196	99.746	99.6095	99.6108	33.4225	33.379	33.4746	33.4804
gray21.512	99.6022	99.643	99.6074	99.6096	33.4608	33.507	33.4588	33.4873
numbers.512	99.6141	99.653	99.6102	99.6092	33.4240	33.388	33.4477	33.4375
ruler.512	99.6120	99.637	99.6092	99.6083	33.4262	33.415	33.4637	33.4617
5.3.01	*99.5931*	99.950	99.6095	99.6101	33.4585	33.508	33.4511	33.4691
5.3.02	99.6128	99.982	99.6095	99.6090	33.4605	*33.514*	33.4536	33.4586
7.2.01	99.6156	99.980	99.6092	99.6097	33.4556	33.487	33.4606	33.4635
testpat.1k	99.6072	99.887	99.6096	99.6098	33.4347	33.453	33.4632	33.4588
Mean	99.6094	99.8131	99.6096	99.6095	33.4476	33.4900	33.4596	33.4612
STD	0.0133	0.1348	0.0018	0.0014	0.0337	0.1240	0.0082	0.0215
Pass/All	26/28	28/28	28/28	28/28	28/28	20/28	28/28	28/28

Italicized data indicates that the test failed

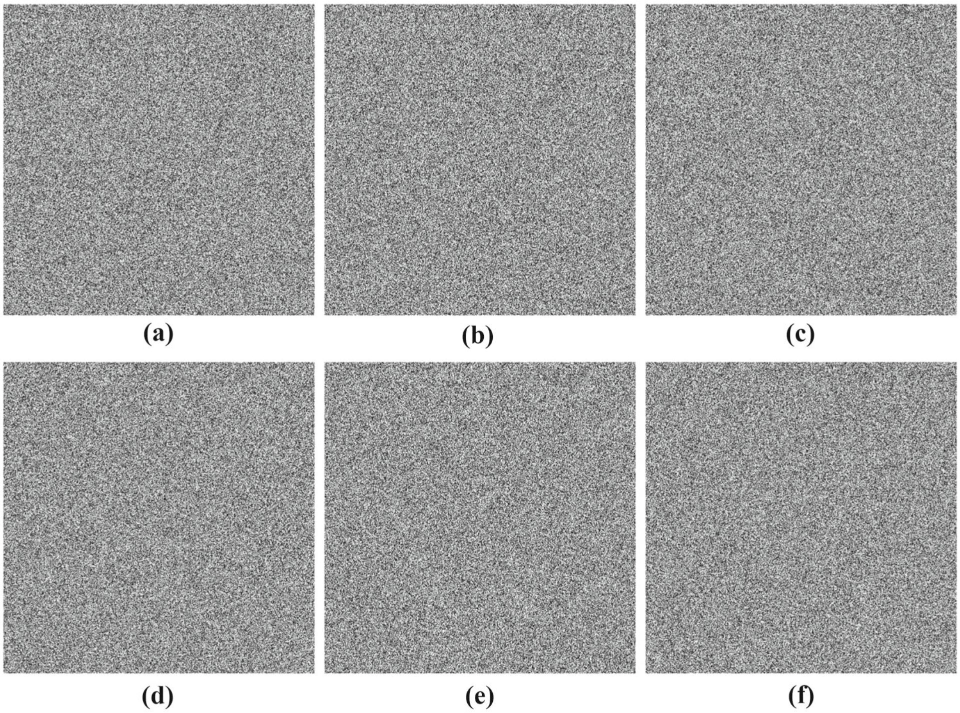


Fig. 20 Encrypted images according to different keys: **a** $P_0(K1)$; **b** $P_1(K2)$; **c** $P_2(K3)$; **d** $P_3(K4)$; **e** $P_4(K5)$; **f** $P_5(K6)$

Table 17 Values of the NPCR and the UACI of encrypted images

Encrypted images	NPCR (%)	UACI (%)
P_0, $P_1(K2)$	99.6090	33.5104
P_0, $P_2(K3)$	99.6021	33.5162
P_0, $P_3(K4)$	99.6067	33.4119
P_0, $P_4(K5)$	99.6010	33.5194
P_0, $P_5(K6)$	99.6396	33.5353

that the contrast of the encrypted image is greatly improved, so the proposed encryption algorithm has better encryption performance.

(3) Homogeneity

The homogeneity is used to describe how close the distribution of elements in the GLCM is to the diagonal of the GLCM [68]. A secure ciphertext image should have small homogeneity. It is defined as

$$Hom = \sum_i \sum_j \frac{p(i,j)}{1 + abs(i-j)} \tag{45}$$

where $p(i,j)$ is the number of GLCM matrices. The range of homogeneity value is [0, 1]. The homogeneity calculation results are shown in Table 19. The results show that the proposed encryption algorithm can effectively reduce the value of homogeneity indicating that the ciphertext image is secure.

4.2.8 Robustness analysis of noise and clipping attacks

When we transmit encrypted images in 5G communication networks, hackers often use some techniques to prevent the information receiver from obtaining the correct decryption information. These techniques include noise attacks and clipping attacks. Therefore, an excellent encryption scheme should not only have the ability to resist clipping attacks and noise attacks, but also have good robustness. In the following experiments, we choose elaine.512 as the test image.

To test the clipping attack, 1/16, 1/8, and 1/4 of the cipher images are permanently damaged, respectively. The damaged cipher images are decrypted separately. The experimental results are shown in Fig. 22. The results show that even if the encrypted image loses 1/4 of the information, the relevant decrypted image can still recover a large amount of original image information. Therefore, the proposed algorithm can effectively resist the clipping attack.

To confirm the robustness of our designed scheme, four kinds of noises, namely salt and pepper noise(SPN), Gaussian noise(GN), speckle noise(SN) and Poisson noise(PN), are added to the encrypted image. The noise intensity(NI) of four kinds of noises is 1%. Then, the correct key is used to decrypt the noisy images, respectively, and the related experimental results are shown in Fig. 23. In addition, according to Eqs. (38) and (39), we can calculate the NPCR and UACI values between different noisy encrypted images and original encrypted images, respectively. At the same time, the PSNR values are also calculated between the relevant decrypted images and the original image, as shown in Table 20. Table 21

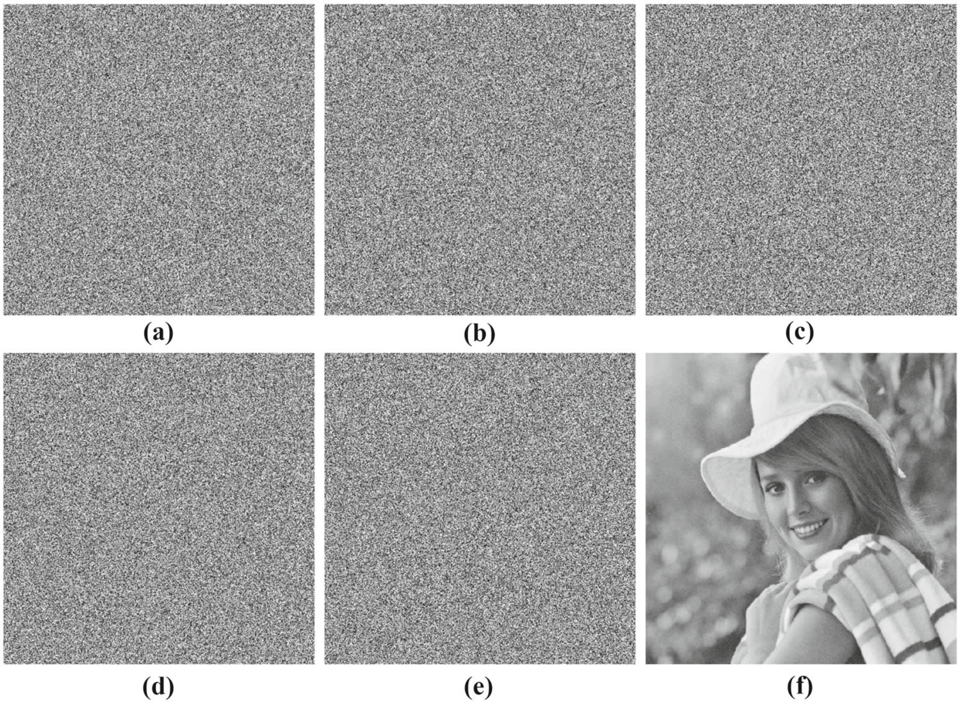


Fig. 21 Decrypted images according to different keys: **a**–**e** $K2$, $K3$, $K4$, $K5$, $K6$; **f** the correct key $K1$

Table 18 NIST test results about three encrypted images

Sub-tests	Encrypted elaine.512 ($p \geq 0.01$)	Encrypted 7.2.01 ($p \geq 0.01$)	ChenTZN ($p \geq 0.01$)
Frequency	0.534146(Pass)	0.534146(Pass)	0.739918(Pass)
BlockFrequency(M =128)	0.350485(Pass)	0.048716(Pass)	0.155209(Pass)
CumulativeSums Forward	0.911413(Pass)	0.162606(Pass)	0.834308(Pass)
CumulativeSums Reverse	0.534146(Pass)	0.213309(Pass)	0.213309(Pass)
Runs	0.350485(Pass)	0.213309(Pass)	0.875539(Pass)
LongestRun	0.739918(Pass)	0.122325(Pass)	0.788728(Pass)
Rank	0.350485(Pass)	0.534146(Pass)	0.949602(Pass)
FFT	0.534146(Pass)	0.122325(Pass)	0.213309(Pass)
NonOverlappingTemplatee (m= 9)*	0.390824(Pass)	0.478983(Pass)	0.513966(Pass)
OverlappingTemplatee(m=9)	0.534146(Pass)	0.739918(Pass)	0.551026(Pass)
Universal	0.739918(Pass)	0.911413(Pass)	0.337162(Pass)
ApproximateEntropy(m=10)	0.122325(Pass)	0.637119(Pass)	0.964295(Pass)
RandomExcursions *	NOT APPLICABLE	NOT APPLICABLE	0.333600(Pass)
RandomExcursionsVariant *	NOT APPLICABLE	NOT APPLICABLE	0.354664(Pass)
Seriale(m=16) p-value1	0.350485(Pass)	0.534146(Pass)	0.311542(Pass)
Seriale(m=16) p-value2	0.534146(Pass)	0.637119(Pass)	0.671779(Pass)
LinearComplexity(M =500)	0.066882(Pass)	0.534146(Pass)	0.337162(Pass)

* The average values of multiple tests

Table 19 Encryption quality analysis results

Test image	Ene			Con			Hom		
	Ref. [67]	Ref. [68]	Our	Ref. [67]	Ref. [68]	Our	Ref. [67]	Ref. [68]	Our
Cameraman	0.01564	0.01564	0.01564	10.6380	10.647	10.54616	0.38503	0.38827	0.3863
All black	0.01568	0.01563	0.01563	10.9829	10.483	10.51529	0.37645	0.38996	0.3886
All white	0.01564	0.01563	0.01564	10.4925	10.614	10.50994	0.38921	0.38842	0.3893

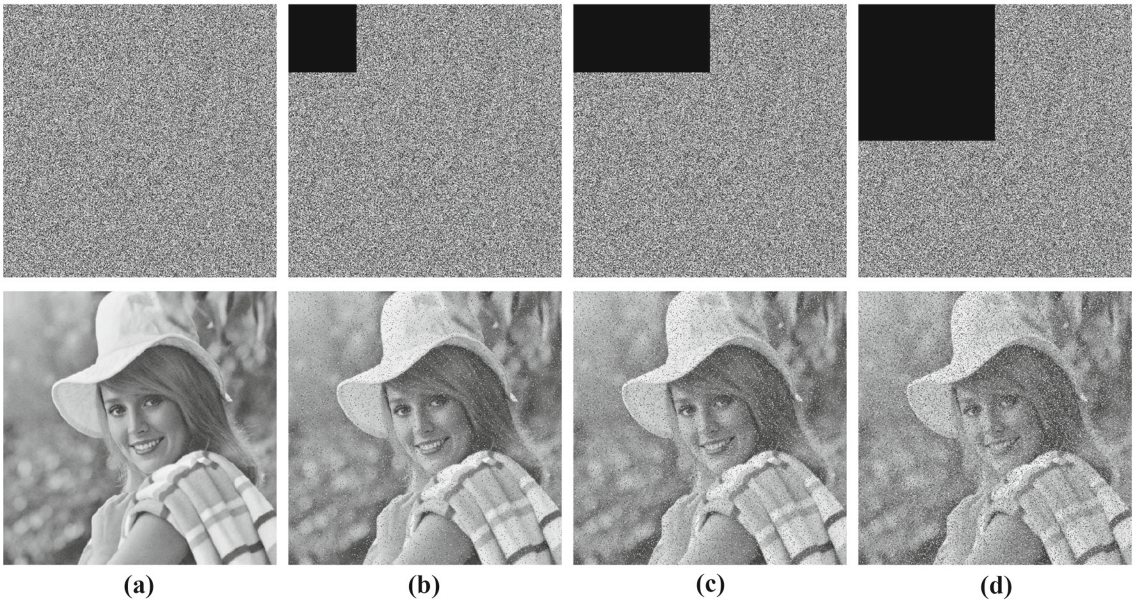


Fig. 22 Simulation results of cropping attacks: **a** the ciphered plain image and the recovered image; **b**–**d** the ciphered images (1/16 loss, 1/8 loss, 1/4 loss) and the corresponding recovered images

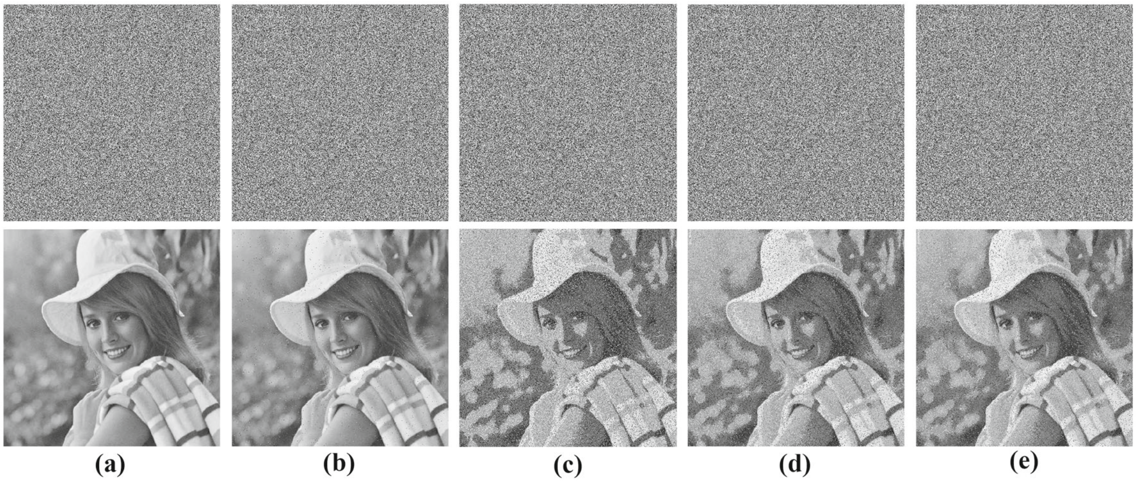


Fig. 23 Simulation results of noise attacks: **a** the ciphered plain image and the recovered image; **b**–**e** the ciphered images (1% salt and pepper noise, 1% Gaussian noise, 1% speckle noise, Poisson noise) and the corresponding recovered images

Table 20 Robustness analysis of four kinds of noises

Test	SPN	GN	SN	PN
NPCR (%)	0.9933	98.0289	93.4501	94.7200
UACI (%)	0.2817	7.4570	4.1038	3.2311
PSNR (dB)	29.32	15.28	17.47	18.34

Table 21 Comparison with other algorithms under different noises

Algorithm	NI (%)	PSNR(dB)		
		SPN	GN	SN
Ref. [69]	0.0003	34.12	31.38	33.29
Ref. [70]	0.0003	28.07	28.00	28.06
Our	0.0003	58.95	33.42	40.62
Ref. [69]	0.0005	34.12	30.24	31.66
Ref. [70]	0.0005	28.07	27.98	28.06
Our	0.0005	58.04	31.46	37.02

shows the comparative experimental results of the anti-noise attack capabilities of different encryption schemes. Although the encrypted image is polluted by various noises, the decrypted images can still recover a lot of original image information. It shows that the proposed scheme can against noise attacks.

Therefore, our scheme not only has the strong ability to resist clipping attacks and noise attacks but also has good robustness.

5 Conclusion

In the paper, a new 1-D compound Sine chaotic system called CSCS is firstly proposed. Using four existing 1-D chaotic maps and the CSCS, then we obtain four new 1-D chaotic maps. Through the analysis of bifurcation diagrams, the comparison of Lyapunov exponents, the comparison of sample entropy and the experiments of NIST statistical test, the four chaotic maps generated by the CSCS show significantly superior chaotic performance. Secondly, we propose a new image encryption scheme known as LSSLCS-IES based on the LSS map and the LCS map produced by the CSCS. Finally, by employing the comparisons of encryption time, key space analyses, the analyses of resistance to statistical attacks, the experiments against differential attacks, the GLCM analysis, and the analyses of resistance to clipping attacks and four types of noise attacks, our scheme has higher security and less time consumption than several other advanced image encryption schemes.

Acknowledgements This research was funded by National Natural Science Foundation of China under Grant No. 61573004, Pilot Project of Fujian Province under Grant No. 2022H0017 and Quanzhou City Science & Technology Program of China under Grant No. 2018C106R.

Declarations

conflict of interest We would like to submit the enclosed manuscript entitled A novel fast image encryption scheme based on a new one-dimensional compound Sine chaotic system, which we wish to be considered for publication in The Visual Computer. No conflict of interest exits in the submission of this manuscript, and manuscript is approved by all authors for publication. I would like to declare on behalf of my co-authors that the work described was original research that has not been published previously, and not under consideration for publication elsewhere, in whole or in part. All the authors listed have approved the manuscript that is enclosed.

References

1. Liu, X., Jia, M., Zhang, X., Lu, W.: A novel multichannel Internet of things based on dynamic spectrum sharing in 5G communication. IEEE Internet Things J. **6**(4), 5962–5970 (2019). https://doi.org/10.1109/JIOT.2018.2847731
2. Wu, Y., Khisti, A., Xiao, C., Caire, G., Wong, K., Gao, X.: A survey of physical layer security techniques for 5G wireless networks and challenges ahead. IEEE J. Sel. Areas Conunun. **36**(4), 679–695 (2018). https://doi.org/10.1109/JSAC.2018.2825560
3. Pak, C., Huang, L.: A new color image encryption using combination of the 1D chaotic map. Signal Process. **138**, 129–137 (2017). https://doi.org/10.1016/j.sigpro.2017.03.011
4. Liu, H., Wang, X.: Color image encryption based on one-time keys and robust chaotic maps. Comput. Math. Appl. **59**(10), 3320–3327 (2010). https://doi.org/10.1016/j.camwa.2010.03.017
5. Liu, H., Wang, X., Kadir, A.: Image encryption using DNA complementary rule and chaotic maps. Appl. Soft Comput. **12**(5), 1457–1466 (2012). https://doi.org/10.1016/j.asoc.2012.01.016
6. Wang, X., Zhang, Y., Bao, X.: A novel chaotic image encryption scheme using DNA sequence operations. Opt. Laser. Eng. **73**, 53–61 (2015). https://doi.org/10.1016/j.optlaseng.2015.03.022
7. Liu, H., Wang, X.: Color image encryption using spatial bit-level permutation and high-dimension chaotic system. Opt. Commun. **284**(16–17), 3895–3903 (2011). https://doi.org/10.1016/j.optcom.2011.04.001
8. Wang, X., Yang, L., Liu, R., Kadir, A.: A chaotic image encryption algorithm based on perceptron model. Nonlinear Dyn. **62**(3), 615–621 (2010). https://doi.org/10.1007/s11071-010-9749-8
9. Wang, X., Liu, L., Zhang, Y.: A novel chaotic block image encryption algorithm based on dynamic random growth technique. Opt. Laser. Eng. **66**, 10–18 (2015). https://doi.org/10.1016/j.optlaseng.2014.08.005
10. Fang, P., Liu, H., Wu, C., Liu, M.: A survey of image encryption algorithms based on chaotic system. Vis. Comput. (2022). https://doi.org/10.1007/s00371-022-02459-5
11. Talhaoui, M.Z., Wang, X., Talhaoui, A.: A new one-dimensional chaotic map and its application in a novel permutation-less image encryption scheme. Vis. Comput. **37**, 1757–1768 (2021). https://doi.org/10.1007/s00371-020-01936-z
12. Hua, Z., Zhou, Y.: Image encryption using 2D Logistic-adjusted-Sine map. Inf. Sci. **339**, 237–253 (2016). https://doi.org/10.1016/j.ins.2016.01.017
13. Hua, Z., Zhou, Y., Huang, H.: Cosine-transform-based chaotic system for image encryption. Inf. Sci. **480**, 403–419 (2019). https://doi.org/10.1016/j.ins.2018.12.048
14. Xian, Y., Wang, X.: Fractal sorting matrix and its application on chaotic image encryption. Inf. Sci. **547**, 1154–1169 (2021). https://doi.org/10.1016/j.ins.2020.09.055
15. Wang, X., Liu, P.: Image encryption based on roulette cascaded chaotic system and alienated image library. Vis. Comput. **38**, 763–779 (2022). https://doi.org/10.1007/s00371-020-02048-4
16. Mansouri, A., Wang, X.: Image encryption using shuffled Arnold map and multiple values manipulations. Vis. Comput. **37**, 189–200 (2021). https://doi.org/10.1007/s00371-020-01791-y
17. Xu, J., Mou, J., Liu, J., Hao, J.: The image compression-encryption algorithm based on the compression sensing and fractional-order chaotic system. Vis. Comput. **38**, 1509–1526 (2022). https://doi.org/10.1007/s00371-021-02085-7
18. Wang, X., Feng, L., Zhao, H.: Fast image encryption algorithm based on parallel computing system. Inf. Sci. **486**, 340–358 (2019). https://doi.org/10.1016/j.ins.2019.02.049
19. Wang, X., Yang, J.: A privacy image encryption algorithm based on piecewise coupled map lattice with multi dynamic coupling coefficient. Inf. Sci. **569**, 217–240 (2021). https://doi.org/10.1016/j.ins.2021.04.013
20. Wang, X., Gao, S.: Image encryption algorithm based on the matrix semi-tensor product with a compound secret key produced by a Boolean network. Inf. Sci. **539**, 195–214 (2020). https://doi.org/10.1016/j.ins.2020.06.030
21. Wang, X., Gao, S.: Image encryption algorithm for synchronously updating Boolean networks based on matrix semi-tensor product theory. Inf. Sci. **507**, 16–36 (2020). https://doi.org/10.1016/j.ins.2019.08.041
22. Xian, Y., Wang, X., Teng, L.: Double parameters fractal sorting matrix and its application in image encryption. IEEE Trans. Circuits Syst. Video Technol. **32**(6), 4028–4037 (2022). https://doi.org/10.1109/TCSVT.2021.3108767
23. Xian, Y., Wang, X., Wang, X., Li, Q., Yan, X.: Spiral-transform-based fractal sorting matrix for chaotic image encryption. IEEE Trans. Circuits Syst. I (2022). https://doi.org/10.1109/TCSI.2022.3172116
24. Wang, X., Liu, C., Jiang, D.: A novel triple-image encryption and hiding algorithm based on chaos, compressive sensing and 3D DCT. Inf. Sci. **574**, 505–527 (2021). https://doi.org/10.1016/j.ins.2021.06.032
25. Cao, C., Sun, K., Liu, W.: A novel bit-level image encryption algorithm based on 2D-LICM hyperchaotic map. Signal Process. **143**, 122–133 (2018). https://doi.org/10.1016/j.sigpro.2017.08.020
26. Gan, Z., Chai, X., Han, D., Chen, Y.: A chaotic image encryption algorithm based on 3-D bit-plane permutation. Neural Comput. Appl. **31**(11), 7111–7130 (2019). https://doi.org/10.1007/s00521-018-3541-y
27. Liu, H., Kadir, A.: Gong, P: A fast color image encryption scheme using one-time S-Boxes based on complex chaotic system and random noise. Opt. Commun. **338**, 340–347 (2015). https://doi.org/10.1016/j.optcom.2014.10.021
28. Zhu, H., Dai, L., Liu, Y., Wu, L.: A three-dimensional bit-level image encryption algorithm with Rubik's cube method. Math. Comput. Simulat. **185**, 754–770 (2021). https://doi.org/10.1016/j.matcom.2021.02.009
29. Zhu, H., Zhang, X., Yu, H., Zhao, C., Zhu, Z.: An image encryption algorithm based on compound homogeneous hyper-chaotic system. Nonlinear Dyn. **89**(6), 61–79 (2017). https://doi.org/10.1007/s11071-017-3436-y
30. Alawida, M., Samsudin, A., Teh, J.S., Alkhawaldeh, R.S.: A new hybrid digital chaotic system with applications in image encryption. Signal Process. **160**, 45–58 (2019). https://doi.org/10.1016/j.sigpro.2019.02.016
31. El-Latif, A.A.A., Niu, X.: A hybrid chaotic system and cyclic elliptic curve for image encryption. AEU **67**(2), 136–143 (2013)
32. Wang, G., Yuan, F.: Cascade chaos and its dynamic characteristics. Acta Phys. Sin. **62**(2), 20506 (2013). https://doi.org/10.7498/aps.62.020506
33. Chen, Z., Yuan, X., Yuan, Y., Iu, H., Fernando, T.: Parameter identification of chaotic and hyper-chaotic systems using synchronization-based parameter observer. IEEE Trans. Circuits Syst. I **63**(9), 1464–1475 (2016). https://doi.org/10.1109/TCSI.2016.2573283
34. Stöckmann, H.-J.: Quantum Chaos: An Introduction. Cambridge University Press, Cambridge (2007)

35. Liu, S., Li, C., Hu, Q.: Cryptanalyzing two image encryption algorithms based on a first-order time-delay system. IEEE MultiMedia (2021). https://doi.org/10.1109/MMUL.2021.3114589
36. Xie, E.Y., Li, C., Yu, S., Lü, J.: On the cryptanalysis of Fridrich's chaotic image encryption scheme. Signal Process. **132**, 150–154 (2017). https://doi.org/10.1016/j.sigpro.2016.10.002
37. Zeraoulia, E., Sprott, J.C.: Robust Chaos and Its Applications. World Scientific, Singapore (2011)
38. Shen, C., Yu, S., Lü, J., Chen, G.: Designing hyperchaotic systems with any desired number of positive lyapunov exponents via a simple model. IEEE Trans. Circuits Syst. I **61**(8), 2380–2389 (2014). https://doi.org/10.1109/TCSI.2014.2304655
39. Richman, J.S., Moorman, J.R.: Physiological time-series analysis using approximate entropy and sample entropy. Am. J. Physiol. Heart Circ. Physiol. **278**, H2039–H2049 (2000). https://doi.org/10.1152/ajpheart.2000.278.6.H2039
40. Bassham, L.E. III, et al.: SP 800-22 Rev. 1A. A statistical test suite for random and pseudorandom number generators for cryptographic applications. Technical report, National Institute of Standards (2010)
41. Grassberger, P., Procaccia, I.: Estimation of the Kolmogorov entropy from a chaotic signal. Phys. Rev. A **278**, 2591–2593 (1983). https://doi.org/10.1103/PhysRevA.28.2591
42. Zhu, H., Zhao, Y., Song, Y.: 2D Logistic–Modulated–Sine–Coupling–Logistic chaotic map for image encryption. IEEE Access **7**, 14081–14098 (2019). https://doi.org/10.1109/ACCESS.2019.2893538
43. Pincus, S.M.: Approximate entropy as a measure of system complexity. Proc. Natl. Acad. Sci. U.S.A. **88**, 2297–2301 (1991)
44. Pincus, S.M.: Approximate entropy(ApEn) as a complexity measure. Chaos **5**, 110–117 (1995). https://doi.org/10.1063/1.166092
45. Eckmann, J.P., Ruelle, D.: Ergodic theory of chaos and strange attractors. Rev. Modern Phys. **57**, 617–654 (1985). https://doi.org/10.1103/RevModPhys.57.617
46. Gong, L., Deng, C., Pan, S., Zhou, N.: Image compression-encryption algorithms by combining hyper-chaotic system with discrete fractional random transform. Opt. Laser Technol. **103**, 48–58 (2018). https://doi.org/10.1016/j.optlastec.2018.01.007
47. Wang, X., Zhang, J., Cao, G.: An image encryption algorithm based on Zigzag transform and LL compound chaotic system. Opt. Laser Technol. **119**, 105581 (2019). https://doi.org/10.1016/j.optlastec.2019.105581
48. Diaconu, A.-V.: Circular inter-intra pixels bit-level permutation and chaos-based image encryption. Inf. Sci. (Ny) **355**, 314–327 (2016). https://doi.org/10.1016/j.sigpro.2017.08.020
49. Xu, L., Li, Z., Li, J., Hua, W.: A novel bit-level image encryption algorithm based on chaotic maps. Opt. Lasers Eng. **78**, 17–25 (2016). https://doi.org/10.1016/j.optlaseng.2015.09.007
50. Chai, X., Chen, Y., Broyde, L.: A novel chaos-based image encryption algorithm using DNA sequence operations. Opt. Lasers Eng. **88**, 197–213 (2017). https://doi.org/10.1016/j.optlaseng.2016.08.009
51. Ping, P., Xu, F., Mao, Y., Wang, Z.: Designing permutation-substitution image encryption networks with Henon map. Neurocomputing **283**, 53–63 (2018). https://doi.org/10.1016/j.neucom.2017.12.048
52. Alawida, M., Teh, J.S., Samsudin, A., Alshoura, W.H.: An image encryption scheme based on hybridizing digital chaos and finite state machine. Signal Process. **164**, 249–266 (2019). https://doi.org/10.1016/j.sigpro.2019.06.013
53. Alvarez, G., Li, S.: Some basic cryptographic requirements for chaos-based cryptosystems. Int. J. Bifurcat. Chaos **16**(8), 2129–2151 (2006). https://doi.org/10.1142/S0218127406015970
54. Yap, W., Phan, R.C.-W., Goi, B., Yau, W., Heng, S.: On the effective subkey space of some image encryption algorithms using external key. Vis. Commun. Image R. **40**, 51–57 (2016). https://doi.org/10.1016/j.jvcir.2016.06.005
55. Wu, Y., Zhou, Y., Saveriades, G., Agaian, S., Noonan, J.P., Natarajan, P.: Local Shannon entropy measure with statistical tests for image randomness. Inf. Sci. **222**, 323–342 (2013). https://doi.org/10.1016/j.ins.2012.07.049
56. Mansouri, A., Wang, X.Y.: A novel one-dimensional sine powered chaotic map and its application in a new image encryption scheme. Inf. Sci. **520**, 46–62 (2020). https://doi.org/10.1016/j.ins.2020.02.008
57. Wang, X., Liu, P.: A new full chaos coupled mapping lattice and its application in privacy image encryption. IEEE Trans. Circuits Syst. I **69**(3), 1291–1301 (2022). https://doi.org/10.1109/TCSI.2021.3133318
58. Zhang, Y., Wang, X.: A symmetric image encryption algorithm based on mixed linear nonlinear coupled map lattice. Inf. Sci. **273**, 329–351 (2014). https://doi.org/10.1016/j.ins.2014.02.156
59. Zhou, Y., Bao, L., Chen, C.: Image encryption using a new parametric switching chaotic system. Signal Process. **93**(11), 3039–3052 (2013). https://doi.org/10.1016/j.sigpro.2013.04.021
60. Chen, G., Mao, Y., Chui, C.K.: A symmetric image encryption scheme based on 3D chaotic cat map. Chaos. Soliton. Fract. **21**, 749–761 (2004). https://doi.org/10.1016/j.chaos.2003.12.022
61. Wu, Y., Noonan, J. P., Agaian, S.: NPCR and UACI randomness tests for image encryption. Cyber J. Multidisci. J. Sci. Technol. J. Select. Areas Telecommun. (JSAT), pp. 31–38(2011)
62. Zhang, Y., Wang, X.: A new image encryption algorithm based on non-adjacent coupled map lattices. Appl. Soft Comput. **26**, 10–20 (2015). https://doi.org/10.1016/j.asoc.2014.09.039
63. Chen, Y., Xie, S., Zhang, J.: A hybrid domain image encryption algorithm based on improved Henon map. Entropy **24**(2), 287 (2022). https://doi.org/10.3390/e24020287
64. Liu, Q., Zhu, C., Deng, X.: An efficient image encryption scheme based on the LSS chaotic map and single S-Box. IEEE Access **8**, 25664–256781291 (2020). https://doi.org/10.1109/ACCESS.2020.2970806
65. Wang, X., Teng, L., Qin, X.: A novel colour image encryption algorithm based on chaos. Signal Process. **92**(4), 1101–1108 (2012). https://doi.org/10.1016/j.sigpro.2011.10.023
66. Khan, J.S., Ahmad, J., Ahmed, S.S., Siddiqa, H.A., Abbasi, S.F., Kayhan, S.K.: DNA key based visual chaotic image encryption. J. Intell. Fuzzy Syst. **37**, 2549–2561 (2019). https://doi.org/10.3233/JIFS-182778
67. Wang, X., Liu, P.: A new image encryption scheme based on a novel one-dimensional chaotic system. IEEE Access **8**, 174463–174479 (2020). https://doi.org/10.1109/ACCESS.2020.3024869
68. Zhu, S., Zhu, C.: Plaintext-related image encryption algorithm based on block structure and five-dimensional chaotic map. IEEE Access **7**, 147106–147118 (2019). https://doi.org/10.1109/ACCESS.2019.2946208
69. Liu, L., Jiang, D., Wang, X., Zhang, L., Rong, X.: A dynamic triple-image encryption scheme based on chaos. S-Box and image compressing. IEEE Access. **8**, 210382–210399 (2020). https://doi.org/10.1109/ACCESS.2020.3039891
70. Zhu, L., Jiang, D., Ni, J., Wang, X., Rong, X., Ahmad, M., Chen, Y.: A stable meaningful image encryption scheme using the newly-designed 2D discrete fractional-order chaotic map and Bayesian compressive sensing. Signal Process. **195**, 108489 (2022). https://doi.org/10.1016/j.sigpro.2022.108489

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Jianeng Tang received the BS.c. degree in electronic information science and technology from Xijiang Normal University, China, 2006; the MS.c. degree in circuits and systems from Ningxia University, China, 2009; the Ph.D. degree in information and communication engineering from Southeast University, China, 2012. He is currently an associate professor at College of Engineering, Huaqiao University, China. His research interests include image encryption, RF circuit design, complex network synchronization, and chaos synchronization and control. He has published over 30 papers in journals and conferences.

Feng Zhang received the BS.c. degree in applied physics from University of Electronic Science and Technology of China, China, 2007. He is currently a deputy general manager of Fujian MM Electronics Co., Ltd. His research interests include image encryption and RF circuit design.

Hui Ni received the BS.c. degree in project management from Fuzhou University, China, 2014. He is currently a deputy general manager of Fujian MM Electronics Co., Ltd.. His research interests include image encryption and RF circuit design.