symmetry

MDPI

Article

Robust Image Encryption Exploiting 2D Hyper-Chaos, Fractal Sierpiński Carpet Confusion, and Cascaded Diffusion

Zeyu Zhang [1], Wenqiang Zhang [2,*], Mingxu Wang [3], Na Ren [2], Peizhen Zhang [2] and Yiting Lin [4,5]

[1] Shenyang Huanke Technology Co., Ltd., Shenyang 110032, China; academicaffairs@sie.edu.cn
[2] Shenyang Institute of Engineering, Shenyang 110136, China; renna@sie.edu.cn (N.R.); zhangpz@sie.edu.cn (P.Z.)
[3] College of Information Engineering, Dalian Ocean University, Dalian 116023, China; 15041119328@163.com
[4] University of Electronic Science and Technology of China Zhongshan Institute, Zhongshan 528406, China; yitinglin@ieee.org
[5] University of Electronic Science and Technology of China, Chengdu 611731, China
* Correspondence: sydzwq@163.com

Abstract

With the rapid growth of digital image transmission, ensuring data security has become increasingly important. However, existing chaos-based image encryption algorithms often suffer from insufficient chaotic randomness and weak integration between chaotic dynamics and encryption mechanisms. To address these issues, a novel image encryption scheme based on a two-dimensional hyperbolic–exponential Sine–Logistic map (2D-HESLM) is proposed. A Sierpiński carpet-inspired scrambling strategy and a cascaded diffusion mechanism are designed to enhance permutation and diffusion performance based on the 2D-HESLM. The experimental results show that the information entropy value is 7.9980, while NPCR and UACI are approximately averaged 99.6147% and 33.4672%, respectively, with correlation coefficients close to zero. These results demonstrate the effectiveness and security of the proposed scheme.

Keywords: image encryption; two-dimensional chaotic map; Sierpiński carpet; permutation–diffusion; fractal-based encryption

1. Introduction

In the context of rapid data proliferation, the large volume of information and the easy-to-understand nature of digital images have enabled their extensive transmission over the internet. However, such widespread transmission and open access inevitably introduce significant security risks, which may lead to image data leakage and serious losses. Therefore, enhancing the security of image data transmission has become increasingly important [1,2]. In this context, image encryption is widely regarded as an effective strategy that protects data by transforming it into unrecognizable content [3–5].

To realize effective image encryption, extensive efforts have been devoted to improving the performance and security of image encryption algorithms (IEAs) in recent years. Various studies have investigated IEAs from different perspectives within the permutation–diffusion framework, including transform and encoding methods [6–9], algebraic constructions [10,11], neural and memristive models [12–17], and chaotic maps (CMs) [15,18–25]. Among these approaches, CM-based IEAs have attracted considerable attention due to their high sensitivity to initial conditions and strong pseudo-random characteristics. For instance, Yan et al. [26] developed an IEA based on a three-dimensional

Check for updates

Academic Editor: Shuang Xu

Received: 11 March 2026
Revised: 30 March 2026
Accepted: 6 April 2026
Published: 10 April 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

Hamiltonian CM combined with chaotic periodic switching functions. Zhang et al. [27] introduced an IEA using a hyper-chaotic CM. Liu et al. [28] proposed a four-dimensional memristive hyperchaotic CM integrated with DNA-based fractal and cross-diffusion techniques, achieving high key sensitivity. In [29], Zhang et al. improved facial feature index enhancement by combining a four-dimensional Chen hyperchaotic CM with circular layered scrambling and bit-level DNA encoding, resulting in near-optimal entropy performance. Overall, the effectiveness of these IEAs is influenced by both the quality of chaotic dynamics and the design of the encryption mechanism.

However, existing CM-based IEAs still suffer from several limitations. Many CMs exhibit insufficient randomness and stability, which may lead to degraded pseudo-random sequences and potential security vulnerabilities. Meanwhile, the encryption process is often not well integrated with chaotic dynamics, and the lack of plaintext-related design further weakens resistance to statistical and differential attacks. These limitations significantly restrict the security performance of existing IEAs. As a result, existing IEAs cannot simultaneously guarantee high-quality pseudo-randomness and effective utilization of chaotic dynamics in the encryption process. More importantly, existing IEAs still lack a unified framework that can simultaneously enhance chaotic dynamics and effectively embed them into the encryption process. Therefore, it is important to investigate how to construct CMs with enhanced dynamical properties and how to design encryption processes that can effectively exploit these properties. In particular, how to achieve a tighter integration between chaotic dynamics and encryption operations remains a key issue.

To address these issues, this work focuses on constructing CMs with enhanced dynamical properties and designing encryption processes that effectively improve image data protection. From the perspective of CM modulation, a 2D-HESLM is constructed to enlarge the chaotic region and enhance randomness. A hash-based parameter initialization with multi-parameter coupling is adopted to expand the key space and improve key sensitivity. From the perspective of IEA design, enhancing the effectiveness of the permutation stage is essential for disrupting spatial correlations. Fractal-based structures, such as the Sierpiński carpet (SC), provide a natural framework for hierarchical and structured scrambling due to their self-similarity and recursive partitioning. These properties enable more flexible and fine-grained pixel rearrangement, thereby improving permutation effectiveness. A fractal-inspired scrambling combined with cascaded diffusion is designed to strengthen confusion and diffusion, effectively reducing statistical correlations. These designs are intended to enhance both the dynamical quality of chaotic sequences and their effectiveness in the encryption process, thereby improving the overall security of the IEA.

The main contributions of this work are summarized as follows:

1. A two-dimensional hyperbolic–exponential Sine–Logistic map (2D-HESLM) is constructed to improve chaotic behavior and enlarge the effective chaotic range.
2. A block scrambling strategy inspired by the SC is designed to achieve more effective permutation and enhance spatial complexity.
3. A cascaded diffusion mechanism is introduced, in which each pixel depends on both the chaotic sequence and its preceding pixel, thereby strengthening resistance to differential attacks.

The remainder of this paper is organized as follows. Section 2 introduces the 2D-HESLM and analyzes its dynamic characteristics. Section 3 presents the SC-based permutation and cascaded diffusion mechanisms. Section 4 describes the process of the proposed scheme. Section 5 provides security analysis and experimental results. Finally, Section 6 concludes this paper.

2. The 2D-HESLM

2.1. Construction of the 2D-HESLM

To make CM-driven IEAs more robust, it is reasonable to combine higher-dimensional CMs endowed with rich dynamics, such as the integration of seed map dimensions into CMs. In this section, we construct the 2D-HESLM based on hyperbolic and exponential functions in the following steps:

Step 1: We chose the hyperbolic functions $\sinh^{-1}(\cdot)$ and $\cosh^{-1}(\cdot)$ as the core basis. These hyperbolic functions stem from the inherent nonlinear properties of hyperbolic functions, such as the derivative $1/\sqrt{y^2+1}$ of $\sinh^{-1}(\cdot)$, which varies based on the input. However, these functions have natural flaws: as the input value increases, the gradient significantly decays, ($\lim_{y\to\infty}(d(\sinh^{-1}(\cdot))/dy) \to 0$). This asymptotic saturation is highly likely to cause trajectories that converge to trivial fixed points in dynamical systems. To overcome this weakness, we made two key modifications: In $\cosh^{-1}(|x|+1)$, $|x|$ can ensure the symmetry of the CM in the phase space and prevent the dynamics from deviating to one side. Moreover, adding 1 forces the input value to always be greater than or equal to 1, which meets the strict domain requirements of $\cosh^{-1}(\cdot)$ and eliminates the risk of CM crashes caused by complex calculations.

Step 2: To further disrupt the regular motion, our strategy is to introduce periodic folding into the y variable: $\mathrm{mod}(y, 2\pi)$ folds the value of y into the $[0, 2\pi)$ interval, and then performs infinite stretching and folding transformations through the $\sin(\cdot)$ function. Next, we use this folded signal for phase perturbation: $h\pi/\sin(\widetilde{y})$. The $\sin(\widetilde{y})$ dynamically changes the amplitude of the disturbance with the position of y within the folding interval. This non-uniformity is crucial for disrupting potential regular periodic orbits. Meanwhile, to ensure a deep and inseparable interdependence between the x and y variables, we designed a bidirectional coupling term: $\sin(\pi xy + \cosh^{-1}(|x|+1))$. The additional $\cosh^{-1}(|x|+1)$ serves as a phase shift, which not only continues the advantages of domain constraints and symmetry guarantees in the first step but also utilizes its nonlinear properties to further disrupt the symmetry of the CM, significantly enhancing its ergodicity, which allows any small phase space region to be mapped to the entire attractor.

Step 3: We add appropriate constraints to the CM to prevent its trajectory from diverging to infinity or collapsing towards the origin. Therefore, introducing controllable dissipation mechanisms is the key to maintaining bounded singular attractors. We used dissipation terms in exponential decay form: $0.1e^{-0.1x_n}$ and $0.1e^{-0.1y_n}$. The principle is that the derivative of the exponential function provides a negative feedback mechanism. When x (or y) is large, the dissipation term approaches zero, and the CM is mainly dominated by chaotic dynamics. When x (or y) approaches zero, the dissipation term output is about 0.1, which acts as a buffer to prevent the attractor from collapsing towards the origin, thus maintaining its spatial extensibility. This occurs because it can minimize its impact when moving away from the origin, without interfering with the generation of chaos, providing sufficient stability when approaching the origin. The coefficient of 0.1 is the critical equilibrium point. It enables the CM to achieve a balance between the local divergence trend caused by chaos and the overall convergence trend caused by dissipation. This balance ensures that the trajectory neither infinitely spreads nor stops moving, but continues to fold and stretch, ultimately forming bounded, fractal-structured strange attractors. Finally, parameter tuning is the key to locking in the chaotic region: the global scaling factor r controls the amplitude of y, while the phase intensity parameter h works in conjunction with x.

Based on the above steps, the constructed 2D-HESLM is

$$\begin{cases} x_{n+1} = \sin\left(\frac{h\pi}{\sin(\widetilde{y}_n)+c} + \sinh^{-1}(y_n)\right) + 0.1e^{-0.1x_n} \\ y_{n+1} = r\sin(\pi x_n y_n + \cosh^{-1}(|x_n|+1)) + 0.1e^{-0.1y_n} \end{cases}, \quad (1)$$

where r and h denote the control parameters of the system, and $\widetilde{y} = \sin(\mathrm{mod}(y, 2\pi)) + 10^{-6}$. The variables x_n and y_n represent the state variables of the system at the nth iteration, while x_{n+1} and y_{n+1} denote the states at the next iteration. The index $n \in N$ indicates the discrete iteration step, where $n = 0, 1, 2, 3, \ldots$. The initial conditions x_0 and y_0 are determined by the secret key and used to generate the chaotic sequences.

2.2. Analysis of the 2D-HESLM

To comprehensively analyze the dynamical properties of the proposed system, several analyses are conducted, including stability, bifurcation, trajectory, and Lyapunov exponent analyses.

2.2.1. Equilibrium Points and Stability Analysis

Equilibrium points and their stability play a fundamental role in understanding the dynamical behavior of nonlinear systems. For discrete CMs, equilibrium points not only describe the structural characteristics of the phase space but also provide essential insights into local dynamics and trajectory evolution. In particular, unstable equilibrium points are often associated with regions of sensitivity, where trajectories undergo repeated stretching and folding, ultimately leading to the formation of chaotic attractors. Therefore, a systematic investigation of equilibrium points is crucial for revealing the intrinsic dynamical mechanisms of the 2D-HESLM.

Equation (1) contains trigonometric terms, inverse hyperbolic functions, and exponential decay factors. In addition, the coupling term πxy introduces further interaction between the state variables. Therefore, it is necessary to reformulate the problem and employ numerical techniques. Based on Equation (1), the equilibrium points are defined as the states that remain unchanged under the system mapping. This condition can be written as $x_{n+1} = x_n = x$ and $y_{n+1} = y_n = y$. This relation indicates that equilibrium points are fixed points of the discrete CM. To facilitate the analysis, the equilibrium conditions are reformulated as a root-finding problem. Two auxiliary functions are introduced as follows:

$$\begin{cases} F_1(x, y) = f(x, y) - x \\ F_2(x, y) = g(x, y) - y \end{cases}, \tag{2}$$

where $f(x, y)$ and $g(x, y)$ denote the two components of the 2D-HESLM defined previously. Under this formulation, equilibrium points correspond to the solutions of $F_1(x, y) = 0$ and $F_2(x, y) = 0$.

From a geometric perspective, the equations $F_1(x, y) = 0$ and $F_2(x, y) = 0$ represent two implicit curves in the phase plane, and their intersections correspond to the equilibrium points of the 2D-HESLM. Under the parameter settings of $r = 3.59$ and $h = 2$, these curves can be visualized, as shown in Figure 1. The red curve corresponds to $F_1(x, y) = 0$, while the blue curve represents $F_2(x, y) = 0$. It can be clearly observed that multiple intersections exist between the two curves, indicating the presence of multiple equilibrium points. The distribution of these intersections across different regions further suggests that the 2D-HESLM possesses a non-uniform and highly structured phase space.

To determine the equilibrium points quantitatively, numerical methods are employed. Specifically, multiple initial guesses are selected within a prescribed region, and a nonlinear solver is applied to locate the roots of the 2D-HESLM. To ensure reliability, redundant solutions are removed through a filtering process. The resulting equilibrium points are summarized in Table 1.

As shown in Table 1, the 2D-HESLM admits multiple equilibrium points distributed over different regions of the phase space. For instance, points such as $S_1 = (0, 0)$ and $S_1 = (0, -0.036402)$ are located near the origin, whereas points like $S_{11} = (0.648183, 0.898746)$

and $S_{14} = (1.870575, 1.936425)$ lie farther away. This distribution reflects the strong nonlinear interactions within the 2D-HESLM and indicates that distinct local dynamical behaviors may arise in different regions.

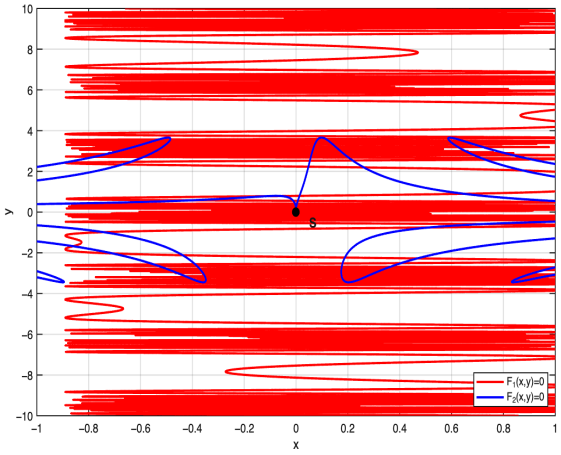


Figure 1. Implicit curves defined by $F_2(x, y) = 0$ and $F_2(x, y) = 0$ and their intersection point under $r = 3.59$ and $h = 2$.

Table 1. Equilibrium points of the 2D-HESLM.

Point	x	y	Point	x	y
S_1	0.000000	0.000000	S_{10}	0.003672	0.420390
S_2	0.000000	−0.036402	S_{11}	0.648183	0.898746
S_3	0.000000	−0.227252	S_{12}	1.048248	1.635316
S_4	0.000000	−0.541756	S_{13}	1.236189	1.91629
S_5	0.000000	−0.805922	S_{14}	1.870575	1.93642
S_6	−0.000123	0.154663	S_{15}	2.821357	0.485213
S_7	−0.010913	0.554920	S_{16}	−1.227876	−0.480155
S_8	−0.114835	0.781044	S_{17}	−1.690537	−1.93813
S_9	−0.404884	0.579063	S_{18}	−2.617985	−1.973529

To further examine the local dynamics near each equilibrium point, the 2D-HESLM is linearized. Let the mapping functions be denoted by $x_{n+1} = f(x, y)$ and $y_{n+1} = g(x, y)$. The Jacobian matrix J at an equilibrium point is given by:

$$J = \begin{bmatrix} \frac{\partial f}{\partial x} - 1 & \frac{\partial f}{\partial y} \\ \frac{\partial g}{\partial x} & \frac{\partial g}{\partial y} - 1 \end{bmatrix}. \tag{3}$$

It should be noted that the $f(x, y)$ involves an auxiliary variable defined as $\widetilde{y} = \sin(\mathrm{mod}(y, 2\pi)) + 10^{-6}$, which introduces an implicit dependency on y. For theoretical analysis and Jacobian derivation, a smooth form $\widetilde{y} = \sin(y)$ is adopted to ensure differentiability. Therefore, the computation of the partial derivative $\partial f / \partial y$ must account for this dependency through the chain rule. Specifically, the derivative is evaluated as:

$$\frac{\partial f}{\partial y} == \frac{\partial f}{\partial \widetilde{y}} \cdot \frac{d\widetilde{y}}{dy} + \frac{\partial f}{\partial \widetilde{y}}\bigg|_{\widetilde{y}}, \tag{4}$$

The J characterizes the linearized behavior of the 2D-HESLM around the equilibrium point. The subtraction of the identity terms arises from the discrete-time formulation. The partial derivatives can be obtained directly from the system equations. The key partial derivatives of the 2D-HESLM can be written as:

$$\begin{cases} \frac{\partial f}{\partial x} = -0.01e^{-0.1x} \\ \frac{\partial f}{\partial y} = \cos\left(\frac{h\pi}{\sin(\widetilde{y})+c} + \sinh^{-1}(y)\right)\left(-\frac{h\pi\cos(\widetilde{y})}{(\sin(\widetilde{y})+c)^2}\cdot\cos(\widetilde{y})\cdot\cos(y) + \frac{1}{\sqrt{y^2+1}}\right) \\ \frac{\partial g}{\partial x} = r\pi xy\cos(\pi xy + \cos^{-1}(|x|+1)\cdot\frac{\mathrm{sign}(x)}{\sqrt{(|x|+1)^2-1}} \\ \frac{\partial g}{\partial y} = r\pi xy\cos(\pi xy + \cos^{-1}(|x|+1)\cdot\frac{\mathrm{sign}(x)}{\sqrt{(|x|+1)^2-1}} - 0.01e^{-0.1y} \end{cases}. \quad (5)$$

After constructing J, its eigenvalues are obtained from $\det(J - \lambda I) = 0$. According to the stability criterion for discrete systems, an equilibrium point is stable only if all eigenvalues satisfy $|\lambda| < 1$. Otherwise, it is unstable. Numerical evaluation shows that each equilibrium point listed in Table 1 has at least one eigenvalue with a magnitude greater than one, indicating that all equilibrium points are unstable.

From a dynamical perspective, this result implies that trajectories cannot remain near any equilibrium point. Instead, they are continuously repelled and move across different regions of the phase space. Moreover, the coexistence of multiple unstable equilibrium points leads to repeated stretching and folding of trajectories, which is a fundamental mechanism underlying chaotic behavior. Therefore, the multi-equilibrium structure and its instability provide a theoretical foundation for the complex dynamics observed in the 2D-HESLM.

2.2.2. Bifurcation Diagram

Bifurcation diagrams (BDs) are utilized as an indispensable mechanism that retains innumerable characteristics that pertain to complex system evolution via means of parametric advancement [30]. BDs can visually depict the long-term behavior of the CM via different control parameters. Figure 2 depicts the BDs of the 2D-HESLM. The blue points represent the state values of the 2D-HESLM after removing the transient iterations under each control parameter. It demonstrates that the 2D-HESLM exhibits a uniform distribution, which means that the 2D-HESLM has enough ergodic properties.

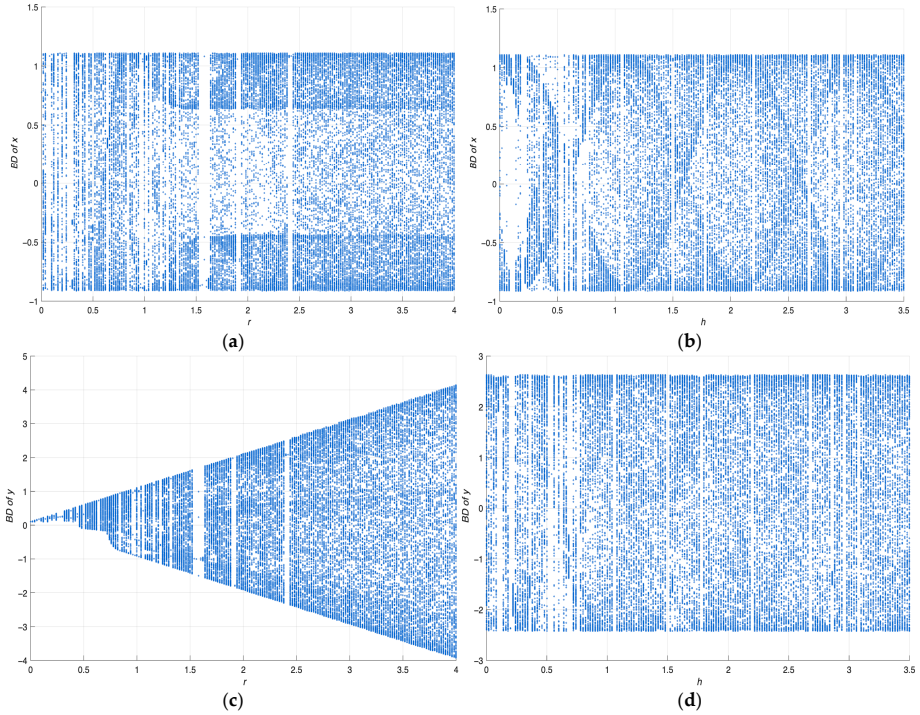


Figure 2. BDs of the 2D-HESLM: (**a**) r and (**b**) h on x_n; (**c**) r and (**d**) h on y_n.

2.3. Trajectory

The dynamic evolution of CMs is a non-repetitive bounded trajectory emanating from initial conditions, whereas unlike periodic oscillation, which follows deterministic cyclic paths and is confined to exact loops, chaotic motions are bounded in space to regions referred to as attraction basins, wherein such chaotic trajectories show topological complexity as they continue to explore different parts of the state domains without converging or becoming periodic. Figure 3 shows that the 2D-HESLM exhibits completely balanced space dispersion similar to CMs [31–33], where the orange points represent the trajectory states after discarding transient iterations.

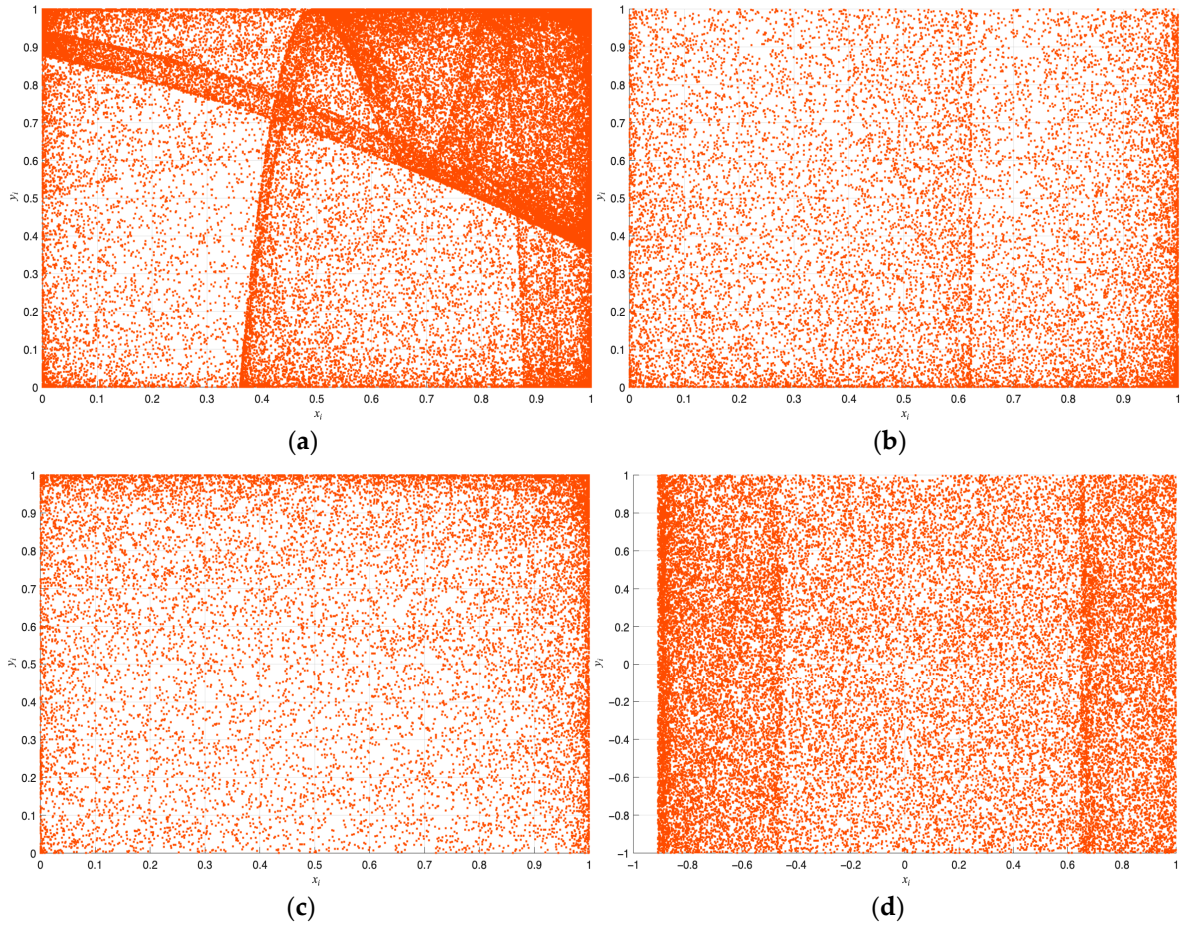


Figure 3. Trajectory plots: (**a**) Sharma [31], (**b**) Gao et al. [32], (**c**) Teng et al. [33], and (**d**) 2D-HESLM.

2.4. Lyapunov Exponent

The Lyapunov exponent (LE) acts as a diagnostic metric to measure the exponential divergence rate between two infinitely close orbits of phase space and is computed by [34]:

$$\lambda = \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln\left|f'(x_i)\right|, \tag{6}$$

where $f'(x_i)$ denotes the derivative of the nonlinear map function f, and x_i represents the CM state at the i-th iteration.

In particular, the maximal LE (MLE) directly determines whether chaos is present. A positive MLE indicates exponential divergence of adjacent system trajectories, causing nearby initial states to evolve into completely different end states over time, and $MLE > 0$ is considered chaotic. When the magnitude of MLE increases, the degree of nonlinear mixing gets bigger too. Under situations where more than one positive MLE can coexist

simultaneously, hyperchaotic regimes with an expansion of high-dimensional divergence can be defined as those with large degrees of high-dimensional divergence, leading to a larger number of state spaces and extending the degree of chaotic entanglement with those standard CMs that only involve a low-dimensional state space. In this work, the Jacobian is further calculated and decomposed into its QR matrix to find the eigenvalues, with the product of the eigenvalues being equal to the Jacobian, allowing us to derive the MLE.

By varying different parameters, the MLE values of the considered CMs are obtained (Figure 4). Figure 4a shows that the first LE remains positive over a wide parameter range, especially for larger values of r, indicating strong chaotic behavior of the 2D-HESLM, while a small non-chaotic region appears in the lower-left area. Warmer colors correspond to larger LE values. Figure 4b shows that the second LE is negative or close to zero in most regions, implying a single dominant expanding direction, with only a few isolated regions exhibiting positive values. Overall, the 2D-HESLM presents a wide chaotic region characterized by $LE > 0$, which is suitable for generating complex pseudo-random sequences. Furthermore, comparisons in Figure 4c,d demonstrate that the proposed map exhibits stronger chaotic behavior than existing CMs.

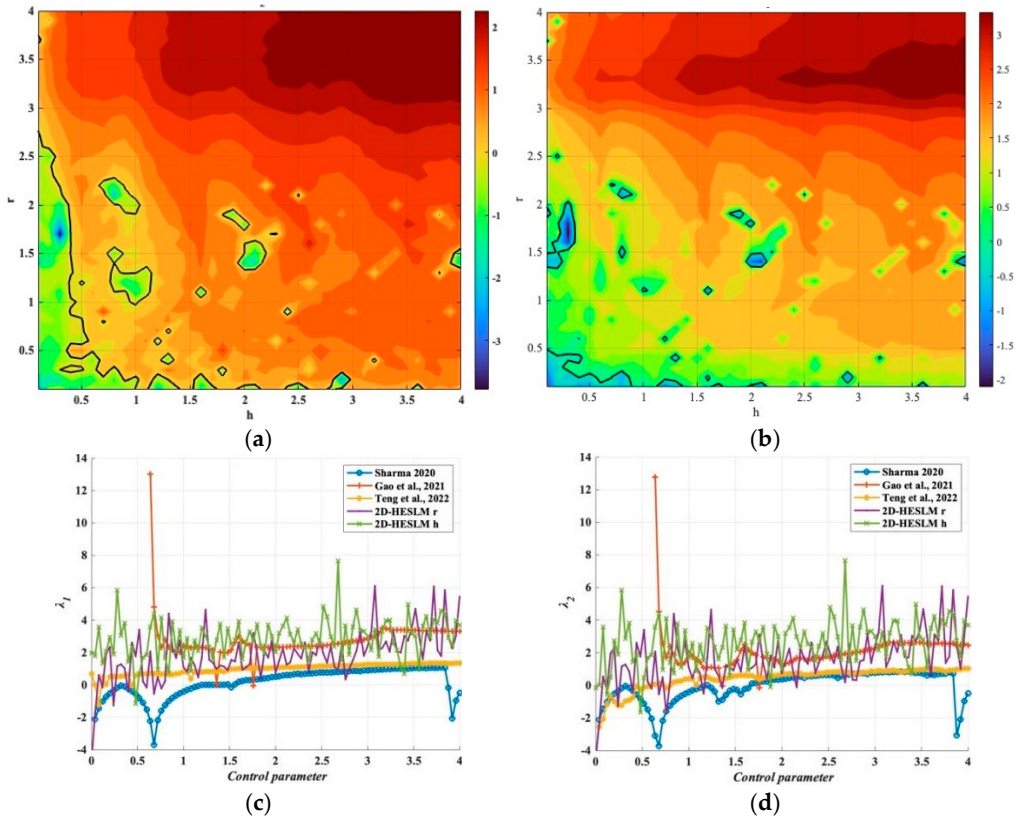


Figure 4. LE analysis of the 2D-HESLM: (**a**) λ_1 and (**b**) λ_2 distributions, and comparative results of (**c**) λ_1 and (**d**) λ_2 with representative CMs [31–33].

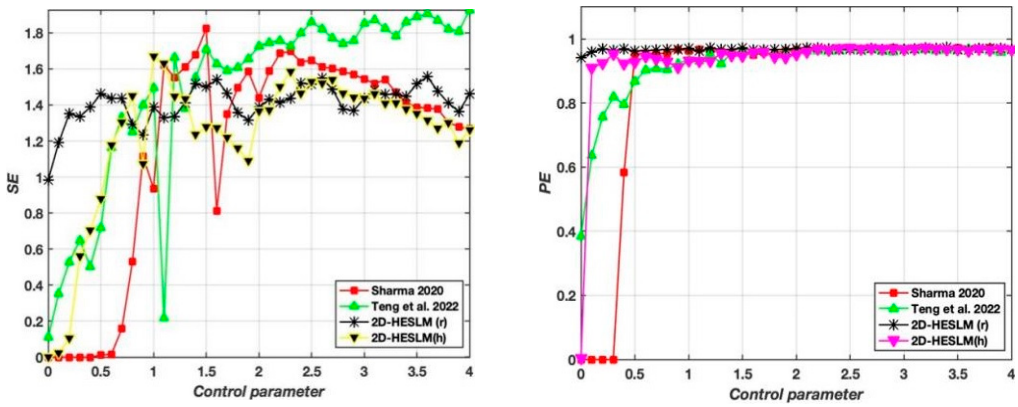
2.5. Sample Entropy

Sample entropy (SE), a measure that characterizes the temporal oscillation pattern of sequential data based on the result of algorithmic analysis of the quantification of the irregularity of the sequence, is directly correlated with system complexity [35]:

$$\mathrm{SE}(m, r, N) = -\log \frac{S}{P}, \tag{7}$$

where m is the template dimension, r means the similarity threshold, and N denotes the temporal sequence length. For the template vectors S and P, the Chebyshev distance

indicates an identical template vector of point distributions relative to a chaotic pattern, if one exists. Note that the SE value may become larger, together with bigger complexity indices. Figure 5a compares the 2D-HESLM with CMs. In contrast to prior CMs, where the SE varies widely from peak values to null measurements within existing methods, the 2D-HESLM exhibits stable values within its specified stability range over various tests. Empirical evidence proves that compared with the traditional CMs, the 2D-HESLM has better temporal pattern-generating performance due to higher structural complexity.



(**a**) (**b**)

Figure 5. Variation in the SE and PE values. (**a**) SE and (**b**) PE [31–33].

2.6. Permutation Entropy

Permutation entropy (PE), as an essential nonlinear index of structural disorder in time series, can evaluate the stochastic intensity of chaotic sequences [36]. Its high values are strongly correlated with sequence complexity; the larger the PE score is, the bigger the complexity configuration and hence the better the performance in recognizing information. In Figure 5b, the PE values confirm that many values remain unstable or quite volatile. In contrast, the 2D-HESLM shows consistent PE oscillations within the stable band. The result suggests that the 2D-HESLM is more stable when it comes to preserving complexity.

2.7. NIST Test

The statistical properties of the generated sequences are further examined using the NIST SP800-22 test suite, which includes 15 tests for evaluating randomness. As tabulated in Table 2, all p-values are greater than 0.01, indicating that the chaotic sequences pass the randomness tests. For the non-overlapping template, random excursions, and random excursions variant, the results are tabulated as pass ratios, which also demonstrate good randomness. Overall, the results demonstrate that the 2D-HESLM achieves strong randomness and can be applied to cryptographic applications.

Table 2. NIST test results of the 2D-HESLM.

Tested Item	Tested Sequence			
	x		y	
Approximate Entropy	0.2543	P *	0.1376	P
Block Frequency	0.4197	P	0.0323	P
Cumulative Sums—Forward	0.1305	P	0.1695	P
Cumulative Sums—Reverse	0.1315	P	0.2549	P
FFT	0.6789	P	0.7701	P
Frequency	0.0760	P	0.2067	P
Linear Complexity	0.6421	P	0.9702	P

Table 2. *Cont.*

Tested Item	Tested Sequence			
	x		y	
Longest Run	0.4282	P	0.2416	P
Non-overlapping Template **	144/148	– ***	148/148	–
Overlapping Template	0.1742	P	0.2717	P
Random Excursions **	8/8	–	8/8	–
Random Excursions Variant **	18/18	–	18/18	–
Rank	0.9988	P	0.4338	P
Runs	0.4960	P	0.7892	P
Serial Test P-value 1	0.3223	P	0.5740	P
Serial Test P-value 2	0.2854	P	0.5192	P
Universal	0.8360	P	0.3244	P

* "P" denotes that the sequence passes the test at the significance level of 0.01, while "F" indicates failure. ** For the non-overlapping template, random excursions, and random excursions variant tests, the results are tabulated as the number of passed subtests over the total number of tests. *** "–" indicates that no separate pass or fail indicator is tabulated for this test. The result is expressed as a pass ratio.

3. SC Theory

3.1. Basic Principles and Recursive Construction

The SC model is an important model of the fractal structure [37]. The basic principle of the model is a square that is evenly divided into nine equal small squares during each iteration, with the middle square removed at each stage and the process repeated recursively for the remaining eight small squares. After numerous iterations, the pattern possesses an infinitely nested pore-like structure, as depicted in Figure 6. Let s denote the iteration level of the fractal construction. For every two iterations, the length of the graph and the number of nodes are $L_{\text{node}} = 3^s$ and $N_{\text{node}} = 3^s$, resulting in Hausdorff dimensions of $d_{\text{H}} = \log 8 / \log 3$. Thus, the process can be strictly represented through recursion relations:

$$F_{\text{n}} = \cup_{i=1}^{8} T_i(F_{n-1}), \tag{8}$$

where T_i denotes an affine transformation applied to the set from the previous iteration, and F_{n} represents the fractal set obtained at the nth iteration. As the number of iterations approaches infinity, the limit set $F_\infty = \cap_{n=1}^{\infty} F_n$ denotes the final fractal set with an infinite self-similar structure. Its properties display basic characteristics of fractal geometry: strict self-similarity, fractional dimension characteristic, and measured dispersion phenomenon.

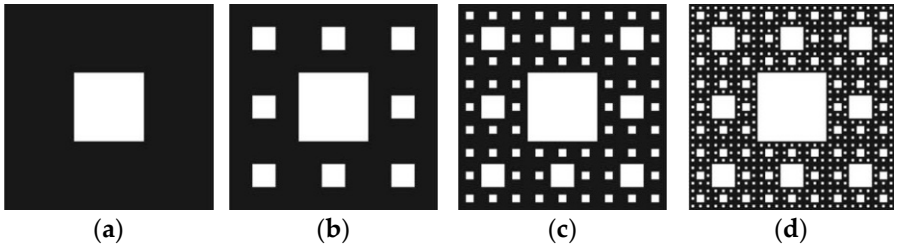


Figure 6. Recursive morphogenesis of the SC fractal with progressive iterative depth: (**a**) primitive lattice (*depth* = 1); (**b**) first-order subdivision (*depth* = 2); (**c**) intermediate hierarchical patterning (*depth* = 3); and (**d**) terminus of self-similar recursion (*depth* = 4).

3.2. Fractal Properties and Motivation for Confusion

Based on the fractal properties of the SC, we can apply it to construct the scrambling process of IEAs. Firstly, we can divide the image into $3^k \times 3^k$ uniform grids according to the previously set level k of recursion, removing the central region's pixel points at each recursion level and executing rotation transformation to the other eight sub-blocks,

thereby achieving affine perturbation of the pixels' matrix. But three problems occur when it is directly applied to IEAs: there are some rigid constraints on the setting of the central block removal operation, which would enable an attacker to figure out the structure of each recursion layer by analysis, allowing them to undertake brute-force attack cracking techniques; fewer pixels in each layer than the previous one causes the pixel distribution to shift irregularly; and uniform segmentation creates structured grids that have predictable layouts, which can be detected through spectrum analysis.

Thus, we adopt this framework to design an SC-based confusion strategy. Its main process is to start with image blocks of size $h \times w$ with starting positions x and y. First, three chaotic sequences are applied to execute row–column shift operations on the blocks. Then, the block is recursively divided into 3×3 non-uniform sub-blocks, and the process is repeated on all sub-blocks. The *depth* controls the recursive hierarchy to ensure termination when the minimum block size (less than 16) or *depth* is exhausted, thereby efficiently generating complex and unpredictable scrambling effects.

4. HESLM-IEA and HESLM-IDA Processes

The proposed scheme consists of two complementary processes, namely the HESLM-IEA and HESLM-based image decryption algorithm (HESLM-IDA). The overall architecture of the HESLM-IEA is illustrated in Figure 7. The framework includes three main modules: key stream initialization, SC-based scrambling, and cascaded diffusion. In the first stage, the SHA-256 hashing and the 2D-HESLM are jointly used to generate plaintext-related chaotic sequences. These sequences are then employed to control the scrambling and diffusion processes. Through recursive block division and row–column operations, spatial correlations are disrupted, while subsequent XOR, modular arithmetic, and cyclic shift operations ensure effective diffusion, thereby achieving secure image encryption.

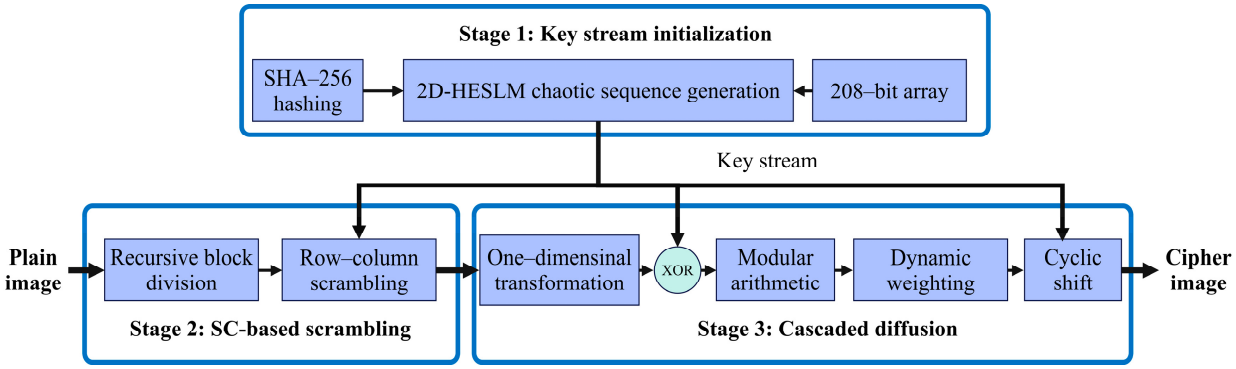


Figure 7. The flowchart of the HESLM-IEA.

4.1. Initialization of Key Stream

The rule is that the key space must be over a minimum dimensional threshold 2^{128} to decode any confidential visual information. In terms of the methodology used for the HESLM-IEA's encrypted parameters, both the hash key K_{HV} of size 2^{256} and pseudo-random bit sequences $K_{\mathrm{B}} = \{bit_1, bit_2, \ldots, bit_{208}\}$ of size 208 are utilized from an overarching scale defined by the SHA-256 hash function, which varies the security components on the basis of variable operation segments. In the equivalent of IEEE 754 double-precision conversion protocols, four relevant parameters are converted into normalized floating-point number equivalents, which will then be acted upon according to Equation (9):

$$x_{\mathrm{init}} = \frac{\sum_{i=1}^{52} bit_i 2^{52-i}}{2^{52}} \tag{9}$$

where $bit_i \in \{0,1\}$ denotes the i-th bit of the pseudo-random binary sequence. The variable x represents the normalized floating-point value derived from the binary sequence. The denominator 2^{52} corresponds to the precision of the mantissa in the IEEE 754 double-precision format, ensuring that $x \in (0,1)$.

The computational process is given by Algorithm 1, where the required mathematical calculations for parameter derivation are formalized.

Algorithm 1. Adjustment of secret keys.

Input : K_B, r, h, x_0, y_0
Output : r', h', x'_0, y'_0
1 : $r' = r + ((\sum_{i=1}^{52} K_B[i] \times 2^{i-1})/2^{52}) \mod (4-r)$
2 : $h' = h + ((\sum_{i=53}^{104} K_B[i] \times 2^{i-53})/2^{52}) \mod (4-h)$
3 : $x'_0 = x_0 + ((\sum_{i=105}^{156} K_B[i] \times 2^{i-105})/2^{52}) \mod (1-x_0)$
4 : $y'_0 = y_0 + ((\sum_{i=157}^{208} K_B[i] \times 2^{i-157})/2^{52}) \mod (1-y_0)$

Moreover, we iterate the 2D-HESLM $IT + 3 \times M \times N$ times to obtain two hyperchaotic signals, X and Y. Then, they are divided into six sub-sequences $CS_i (i = 1, 2, 3, \ldots, 6)$:

$$\begin{cases} CS_1 = floor(X(1{+}HM(1) : M \times N + HM(1)) \times 10^{\alpha}) \text{mod BL} \\ CS_2 = floor(X(M \times N + HM(2) + 1{:}\ 2 \times M \times N + HM(2)) \times 10^{\alpha}) \text{mod BL} \\ CS_3 = floor(X(2 \times M \times N + HM(3) + 1 : 3 \times M \times N + HM(3)) \times 10^{\alpha}) \text{mod BL} \\ CS_4 = floor(Y(1{+}HM(4) : M \times N + HM(4)) \times 10^{\alpha}) \text{mod BL} \\ CS_5 = floor(Y(M \times N + HM(5) + 1{:}\ 2 \times M \times N + HM(5)) \times 10^{\alpha}) \text{mod BL} \\ CS_6 = floor(Y(2 \times M \times N + HM(6) + 1 : 3 \times M \times N + HM(6)) \times 10^{\alpha}) \text{mod BL} \end{cases} \quad (10)$$

where X and Y denote the two chaotic sequences generated by the 2D-HESLM, and IT represents the number of transient iterations discarded to eliminate initial effects. The function $HM(i)$ denotes the i-th element of the hash-derived sequence used for index perturbation. The sequences $CS_i (i = 1, 2, \ldots, 6)$ represent the generated integer sub-sequences used in subsequent encryption operations. The operator $floor(\cdot)$ maps a real value to the largest integer, which is not greater than itself, and mod denotes the modulo operation. $BL = 256$ defines the quantization range of pixel values, and α is a scaling parameter (set to $\alpha = 8$) used to enhance the numerical sensitivity of the chaotic sequences.

4.2. HESLM-IEA Process

This part introduces the process of the HESLM-IEA.

Step 1: We generate the key stream of the HESLM-IEA based on Section 4.1. The original image PI of the dimension $M \times N$ is hashed with SHA-256 and saved as HM. Then, it is converted into an eight-decimal number sequence using Equation (11):

$$HM = hex2dec(K_{HV}(i : i+7)), \quad (11)$$

where the function $hex2dec(\cdot)$ converts a hexadecimal number into its decimal equivalent. The index i indicates the starting position of the substring, and $K_{HV}(i : i+7)$ denotes an 8-digit hexadecimal segment of K_{HV}. The resulting sequence HM consists of decimal values derived from the hash and is used for key generation and parameter initialization in the encryption process.

To run the HESLM-IEA, first, each hexadecimal element HM gets changed into a decimal value; then, we apply the modulus to keep the values bounded $[0, 255]$. Moreover, we obtained the 2D-HESLM's initial parameters (r, h, x_0, and y_0) and six sub-sequences via Equation (9).

Step 2: SC-based recursive scrambling process.

Step 2.1: Read the recursion depth k and three chaotic sentences ss_1, ss_2, and ss_3.

Step 2.2: Perform the block extraction and chaotic sequence positioning. We start from position (1, 1) and define the current rectangular block as the size $h \times w$. The start positions for extracting the sequence are calculated based on the position of the sub-block:

$$offset = (x - 1) \times W + y, \tag{12}$$

where W denotes the width of the image. The variables x and y represent the row and column indices of the current sub-block, respectively:

$$ss_i = ss_i(offset : offset + h \times w - 1), \tag{13}$$

where h and w denote the height and width of the sub-block. The sequences ss_1, ss_2, and ss_3 are chaotic sequences generated from the 2D-HESLM. The variable offset specifies the starting index for sub-sequence extraction.

Step 2.3: The CM-based row–column joint scrambling is presented. We perform a double confusion of rows and columns on the current sub-block. The specific process is as shown in Algorithm 2.

Algorithm 2. Chaos-based row–column shift.

Input: $Block$ **(**$h \times w$**),** ss_1, ss_2, and ss_3
Output: $Block_{\text{cir}}$
1: $row_{\text{Perm}} \leftarrow \text{argsort}(ss_1[1:h])$;
2: $row_{\text{Shift}}(i) \leftarrow \lfloor ss_2[i] \times 10^4 \rfloor \bmod w, \forall i \in [1, h]$;
3: $Block_{\text{cir}} \leftarrow Block[row_{\text{Perm}}, :]$;
4: for each $i = [1, h]$ **do**
5: $Block_{\text{cir}}[i, :] \leftarrow \text{circsift}(Block_{\text{cir}}[i, :], row_{\text{Shift}}[i], 2)$;
6: end for
7: $col_{\text{Perm}} \leftarrow \text{argsort}(ss_2[1:w])$;
8: $col_{\text{Shift}}(j) \leftarrow \lfloor ss_3[h + j] \times 10^4 \rfloor \bmod h, \forall j \in [1, w]$;
9: $Block_{\text{cir}} \leftarrow Block_{\text{cir}}[:, col_{\text{Perm}}]$;
10: for each $j = [1, w]$ **do**
11: $Block_{\text{cir}}[:, j] \leftarrow \text{circsift}(Block_{\text{cir}}[:, j], col_{\text{Shift}}[j], 1)$;
12: end for

Step 2.4: The three-section recursive decomposition is executed. We divide the current sub-block into three parts equally in both height and width:

$$h_{\text{sub}} = \left[\left\lfloor \frac{h}{3} \right\rfloor, \left\lfloor \frac{h}{3} \right\rfloor, h - 2 \times \left\lfloor \frac{h}{3} \right\rfloor \right], \tag{14}$$

$$w_{\text{sub}} = \left[\left\lfloor \frac{w}{3} \right\rfloor, \left\lfloor \frac{w}{3} \right\rfloor, w - 2 \times \left\lfloor \frac{w}{3} \right\rfloor \right], \tag{15}$$

where h and w denote the height and width of the current sub-block, respectively. The vectors h_{sub} and w_{sub} represent the partition sizes along the height and width directions after decomposition. This partition ensures that the sub-block is divided into three integer-sized segments in each dimension.

For the generated 3×3 new sub-blocks, sequentially from top-left to bottom-right, steps 2.1–2.4 are repeated with recursion depth $k - 1$ until the recursion depth $k = 0$ or the sub-block size is smaller than 16×16. Finally, we obtain the confused plain image PI_{Con}. Figure 8 depicts an example of scrambling for a 5×5 matrix, where the matrix is divided into nine blocks when $d = 1$; whereas in the case of $B5$, which is shown alone, the sub-block's scrambled form resulting from row–column confusion at $d = 2$ is presented. Different colors in Figure 8 are used to distinguish sub-blocks generated during

the recursive decomposition process. Each color represents an independent block that undergoes row-column scrambling, illustrating the transformation and reorganization of pixel positions across different stages. The color variations help visualize the structural evolution of the scrambling procedure.

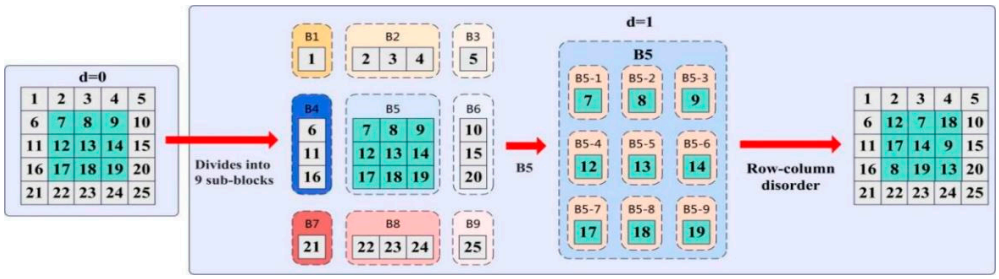


Figure 8. A scrambling example on a 5 × 5 matrix.

Step 3: Cascaded diffusion.

Step 3.1: We convert the scrambled image PI_{Con} into a one-dimensional array img_{Vec} of size $len = M \times N$. Meanwhile, three chaotic sequences CS_4, CS_5, and CS_6 are reshaped into one-dimensional vectors. Based on CS_5, we calculate a dynamic weight vector w_{DWV}:

$$w_{\mathrm{sub}} = \left[\left\lfloor \frac{w_{\mathrm{DWV}}}{3} \right\rfloor, \left\lfloor \frac{w_{\mathrm{DWV}}}{3} \right\rfloor, w_{\mathrm{DWV}} - 2 \times \left\lfloor \frac{w_{\mathrm{DWV}}}{3} \right\rfloor \right], \tag{16}$$

where mod represents the modulo operation, ensuring $w_{\mathrm{DWV}}(i) \in \{1, 3, 5, 7\}$. It is used for subsequent multiplication operations to enhance the nonlinearity of pixel transformation.

Step 3.2: We start the diffusion operation. Firstly, we initialize a state vector SA, whose length is one more than the total number of pixels. All elements are initially set to 0, but the last element SA is set to the last value of CS_6. For each pixel, we obtain the value of its previous pixel. If it is the first pixel, its pre-pixel is set to 0; otherwise, the pre-pixel is taken from SA. Then, the pixel value img_{Vec} is added to CS_4, and the result is taken as modulo 256 to ensure the value is within the range of 0–255. This result is XORed with the previous pixel, generating an intermediate value mix_{IDV}:

$$mix_{\mathrm{IDV}} = SA(i-1) \oplus ((img_{\mathrm{Vec}}(i) + CS_4(i)) \mathrm{mod} 256), \tag{17}$$

where $SA(i-1)$ represents its previous element. The variable $img_{\mathrm{Vec}}(i)$ represents the i-th pixel value. This value is then multiplied by weight w_{DWV}, and again taken as modulo 256, obtaining the mixed pixel value mix_{IDV}:

$$mix_{\mathrm{IDV}} = (mix_{\mathrm{IDV}} \times w_{\mathrm{DWV}}(i)) \mathrm{mod} 256, \tag{18}$$

Step 3.3: We use the chaotic sequence CS_6 to generate a cyclic left shift (V_{CLS}) value and calculate the shift bit number $b_{\mathrm{SBVI}} = CS_6(i) \mathrm{mod} 8$, which ensures that $0 \leq b_{\mathrm{SBVI}} \leq 7$. Then, we apply the V_{CLS}:

$$V_{\mathrm{CLS}}(CS_6(i), b_{\mathrm{SBVI}}) = ((CS_6(i) \ll b_{\mathrm{SBVI}}) | (CS_6(i) \gg (8 - b_{\mathrm{SBVI}}))) \mathrm{mod} 256, \tag{19}$$

The V_{CLS} recombines the high and low bits to produce a pseudo-random value. Then, mix_{IDV} is XORed with V_{CLS} to update the state vector SA:

$$SA(i) = mixed \oplus V_{\mathrm{CLS}}(CS_6(i), b_{\mathrm{SBVI}}), \tag{20}$$

It is worth noting that V_{CLS} also affects the next pixel. If the next pixel is not the last pixel, the V_{CLS} is XORed with the initial value of $SA(i+1)$

$$SA(i+1) = SA(i+1) \oplus V_{\mathrm{CLS}}(CS_6(i), b_{\mathrm{SBVI}}), \tag{21}$$

The whole diffusion process can be denoted as:

$$\begin{cases} s(i) = [(s(i-1) \oplus (P(i) + CS_1(i)) \times w_i]_{256} \oplus V_{\mathrm{CLS}}(CS_3(i), CS_3^*(i+1)) \\ s(i+1) = s(i+1) \oplus V_{\mathrm{CLS}}(CS_3(i), CS_3^*(i+1)) \end{cases}, \quad (22)$$

where $s(i)$ denotes the current state value, and $s(i-1)$ represents its previous state. $P(i)$ denotes the i-th pixel value, and CS_1 and CS_3 are chaotic sequences. The term CS_3^* represents the transformed or shifted version of the chaotic sequence CS_3 used in the V_{CLS} operation.

Step 3.4: We reshape S into the final cipher image EI.

4.3. HESLM-IDA Process

This part introduces the process of the HESLM-IDA, which is the inverse procedure of the HESLM-IEA. All operations are reversible and executed in the exact reverse order of the encryption process.

Step 1: Key stream regeneration: The same key stream is regenerated based on Section 4.1. Specifically, the K_{HV} is converted into the decimal sequence HM using Equation (11). Subsequently, the initial parameters of the 2D-HESLM and the corresponding chaotic sequences $CS_i (i = 1, 2, \ldots, 6)$ are reproduced identically to those in the encryption stage.

Step 2: Inverse cascaded diffusion.

Step 2.1: The cipher image EI is reshaped into a one-dimensional sequence S, and the same chaotic sequences (CS_1, CS_3, and CS_4) and the weight vector w_{DWV} are reconstructed.

Step 2.2: The diffusion process is reversed by processing S in reverse order. Based on the state vector SA, the influence of the V_{CLS} operation, XOR operation, and modular arithmetic are sequentially removed to recover the intermediate mix_{IDV}.

Step 2.3: The effect of the cyclic left shift $V_{\mathrm{CLS}}(CS_3(i), CS_3^*(i+1))$ is eliminated using the inverse shift operation, and the XOR operation with the chaotic sequence CS_1 is reversed to obtain the corresponding pixel contribution.

Step 2.4: The original pixel values $img_{\mathrm{Vec}}(i)$ are reconstructed by reversing the state-dependent update of $SA(i)$, and the recovered one-dimensional sequence is reshaped into the scrambled image PI_{Con}.

Step 3: Inverse SC-based scrambling.

Step 3.1: Using the same recursion depth k and chaotic sequences ss_1, ss_2, and ss_3, the scrambling process is reversed.

Step 3.2: The inverse row–column operations are applied using the same permutation indices and shift values to restore pixel positions within each sub-block.

Step 3.3: The recursive merging is performed in the reverse order of decomposition based on the partition sizes to reconstruct larger blocks.

Step 3.4: The process is repeated until all sub-blocks are merged, and the original plain image PI is finally recovered.

5. Simulation Results and Performance Analysis

To evaluate the performance of the HESLM-IEA, test images of different sizes are considered in this work. All experiments reported in this section were conducted using standard grayscale images with a size of 256 × 256 pixels. The images are selected from the standard USC-SIPI image database (https://sipi.usc.edu/database/database.php, accessed on 13 March 2026). The same set of test images was used across all experiments to avoid result-oriented selection.

5.1. Simulation Result

To visually evaluate the performance of the proposed scheme, representative test images were selected for encryption and decryption experiments, as shown in Figure 9. The encrypted images appear noise-like, with no visible structural information from the original images, indicating strong resistance to visual and statistical attacks. In contrast, the decrypted images are accurately recovered to their original forms without observable distortion, demonstrating the lossless property of the proposed algorithm. These results confirm that the HESLM-IEA effectively obscures visual information while the HESLM-IDA correctly reconstructs the original images, satisfying the fundamental requirements of practical IEAs.

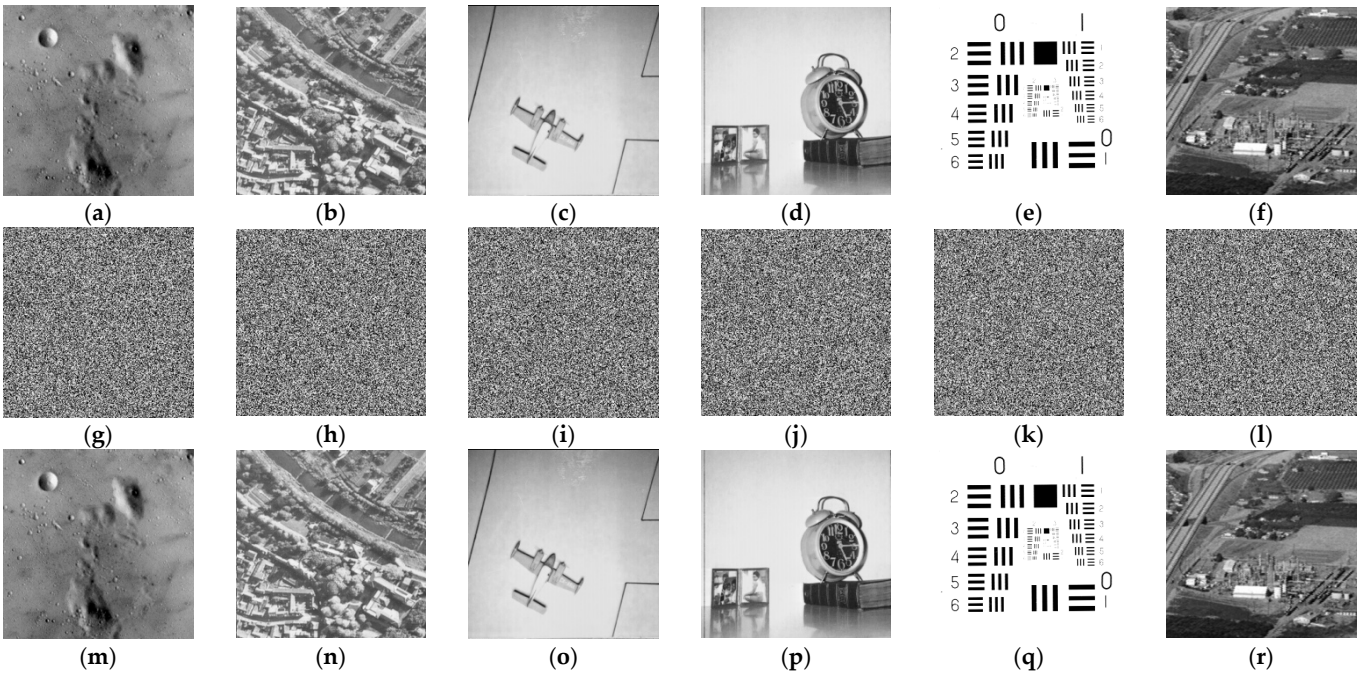


Figure 9. Stimulation results: (**a**–**f**) plain images, (**g**–**l**) cipher images of (**a**–**f**), and (**m**–**r**) decoded images of (**g**–**l**).

5.2. Key Space

The key space is linked with the robustness of security. For the HESLM-IEA, the SHA-256 hashing is utilized to obtain the encryption parameters K_{H}, forming a key space equal to 2^{256}. Meanwhile, a 208-bit value K_{B} is used to adjust the initial values of the 2D-HESLM. The security parameter space derived from this process was confirmed to achieve up to $2^{256} \times 2^{208} \approx 2^{464}$, and far exceeded that of 2^{100}. It ensures heightened ability against brute force attacks.

5.3. Key Sensitivity

A robust IEA should exhibit high sensitivity to slight variations in the secret key. To evaluate this property, a small perturbation of magnitude 10^{-15} is introduced to r' and $HM(1)$, while all other parameters remain unchanged. Figures 10 and 11 show the results under such slight key modifications. From the HESLM-IEA perspective, it can be observed that even an extremely small change in the key leads to significantly different encrypted images, as shown in Figure 10. To further quantify this effect, the histogram of Figure 10d was analyzed. The distribution exhibits a widespread transmission, indicating that pixel differences are randomly distributed over the entire dynamic range. From the HESLM-IDA perspective, when these keys are used in the HESLM-IDA, only the matched key can

decrypt the corresponding image, while those keys that have been slightly modified cannot obtain the corresponding image, as shown in Figure 11. Moreover, the decrypted images obtained from two different modified keys are completely different. This indicates that the keys of the algorithm have a very strong sensitivity.

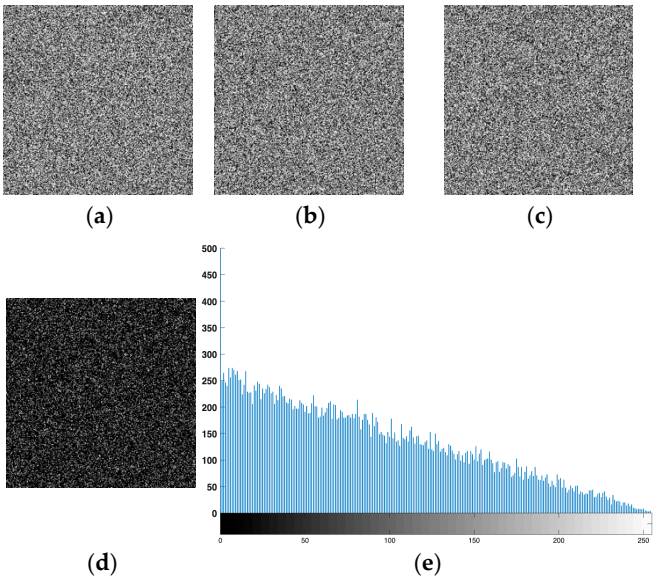


Figure 10. Key sensitivity of the HESLM-IEA: (**a**) the original image "5.1.09"; (**b**,**c**) the encrypted results using r' and $HM(1)$; (**d**) the differential result of (**b**,**c**); and (**e**) the histogram of (**d**).

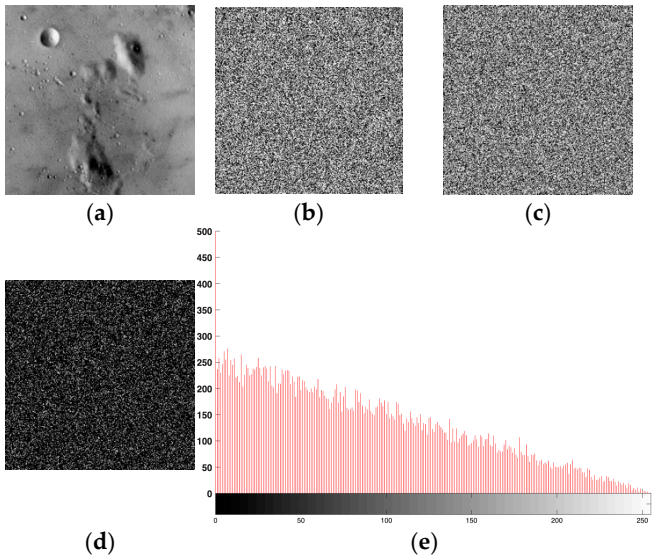
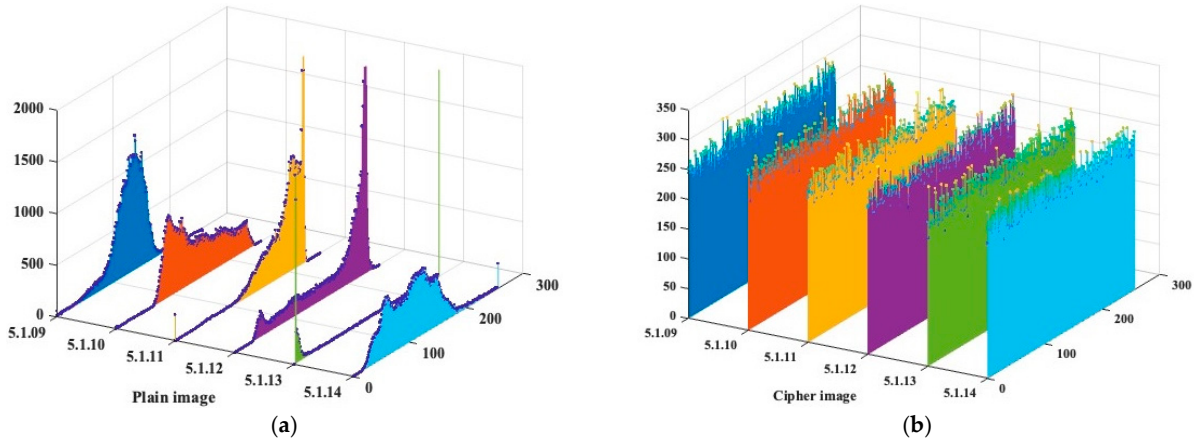


Figure 11. Key sensitivity of the HESLM-IDA: the decrypted image of the cipher image "5.1.09" using (**a**) the original keys, (**b**) r', and (**c**) $HM(1)$; (**d**) the differential result of (**b**,**c**); and (**e**) the histogram of (**d**).

5.4. Histogram

Histograms are commonly used to visually evaluate the distribution of grayscale pixel intensities through frequency counts [30]. Figure 12 compares the histograms of the original and encrypted images. The original images exhibit irregular and uneven distributions, whereas the encrypted images show a nearly uniform histogram. Such uniformity indicates that the statistical characteristics of the original images are effectively concealed in the ciphertext, making it difficult for attackers to extract useful information through histogram-based analysis. Therefore, it can be concluded that the HESLM-IEA effectively removes distinguishable patterns and enhances resistance against statistical attacks.



(**a**) (**b**)

Figure 12. Histograms: (**a**) the original images and (**b**) encrypted images.

Furthermore, quantitative validation using the chi-square test (χ^2) evaluates uniformity with a statistical calculation of the level of adherence to an equal distribution of pixels and is computed as follows:

$$\chi^2 = \sum_{L=0}^{255} \frac{(o_L - e_L)^2}{e_L}, \tag{23}$$

where o_L and e_L stand for the observed and expected pixel counts of gray levels L, respectively. For an 8-bit image, a χ^2 value lower than the critical threshold of 293.2478 (at a significance level of 0.05) indicates that the pixel distribution conforms to a uniform distribution. The results are summarized in Table 3, where "Pass" denotes that the corresponding χ^2 value is below the threshold. It can be observed that all encrypted images satisfy this criterion, while the original images exhibit significantly larger deviations. These results demonstrate that the HESLM-IEA effectively produces uniformly distributed ciphertexts and can resist statistical attacks.

Table 3. χ^2 results.

Test Image	**5.1.09**	**5.1.10**	**5.1.11**	**5.1.12**	**5.1.13**	**5.1.14**
χ^2	224.8672	257.7656	253.0781	247.1797	264.2969	234.5000
Result	Pass	Pass	Pass	Pass	Pass	Pass

5.5. Correlation

The correlation of neighboring pixels on diagonal (D), vertical (V), and horizontal (H) orientations was tested on different images. Correlation values, in which the absolute range is between 0 and 1, gauge how closely two pixels are related to one another, with values near 0 indicating low relation, while 1 depicts close linear correlation. A good IEA can produce a cipher image without any obvious patterns present in the original image. This in turn leads to correlation coefficients near 0 in each orientation [38]. The following formulas represent the calculation of correlation metrics.

$$r_{xy} = \frac{\mathrm{cov}(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{24}$$

$$\mathrm{cov}(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - (E(y)), D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2, E(x) = \frac{1}{N}\sum_{i=1}^{N}x_i, \tag{25}$$

where x and y denote the sequence of N pixels. $\mathrm{cov}(x,y)$ is their covariance. $D(x)$ and $D(y)$ are the variances. $E(x)$ and $E(y)$ are the average values.

In Table 4 and Figure 13, the pixel distribution of encrypted images is statistically uncorrelated, as indicated by the calculated correlation coefficients. Near-zero coefficients in the horizontal, vertical, and diagonal directions demonstrate that the HESLM-IEA effectively removes spatial dependencies between adjacent pixels. As shown in Figure 13, the original image exhibits strong linear correlations in three directions, whereas the encrypted image presents a random scatter distribution without visible structure. In addition, Table 4 shows that the correlation coefficients of the encrypted images are consistently close to zero and lower than those of the compared IEA, indicating stronger decorrelation capability. This property enhances resistance against statistical attacks, as attackers cannot exploit pixel relationships to infer useful information from the ciphertext.

Table 4. Correlation coefficient results.

Test Image	HESLM-IEA						Ref. [39]		
	Original			Encrypted			Encrypted		
	H	V	D	H	V	D	H	V	D
5.1.09	0.9713	0.9481	0.9589	−0.0024	−0.0006	0.0003	0.0003	−0.0053	−0.0011
5.1.10	0.8930	0.9053	0.9015	0.0034	−0.0015	0.0017	0.0053	−0.0041	−0.005
5.1.11	0.9460	0.9558	0.9644	−0.0022	0.0033	0.0021	0.0045	0.0004	0.0025
5.1.12	0.9591	0.9686	0.9548	0.0018	0.0024	−0.0028	0.0004	0.0014	−0.0019
5.1.13	0.8890	0.8765	0.8688	−0.0012	0.0005	0.0005	−0.0007	0.0068	−0.0074
5.1.14	0.9540	0.9518	0.9438	0.0006	0.0034	0.0026	−0.005	−0.005	0.0042

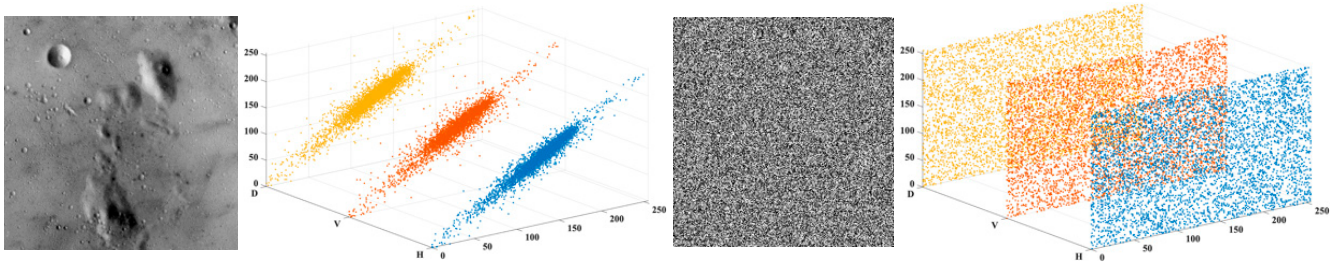


Figure 13. Correlation results: (**a**) the plain image "5.1.09"; (**b**) the correlation distribution of adjacent pixels in three directions of (**a**); (**c**) the encrypted image of (**a**); and (**d**) the correlation distribution of adjacent pixels in three directions of (**c**).

5.6. Information Entropy

Shannon information entropy (SIE) can measure the level of disorder contained in the color channels of cipher images (see [40]). For grayscale images, SIE is assessed through Equation (26):

$$H(x) = -\sum_{i=1}^{L} p(x_i) \log_2 p(x_i), \tag{26}$$

where $p(x_i)$ denotes the probability of the occurrence of gray-level x_i among L possible values. For an 8-bit image, the theoretical maximum entropy is 8, and values closer to 8 indicate higher randomness.

As shown in Table 5, the entropy values of the encrypted images are all close to 7.9980, which are very close to the theoretical maximum and comparable to or slightly better than those of existing IEAs. This demonstrates that the HESLM-IEA produces highly random ciphertexts and effectively conceals the statistical characteristics of the original images.

Table 5. SIE values.

Test Image	Original Image	Encrypted Image				
		Ref. [39]	Ref. [41]	Ref. [42]	Ref. [43]	HESLM-IEA
5.1.09	6.7093	7.9971	-	7.9973	7.9972	7.9975
5.1.10	7.3118	7.9972	7.9971	7.9975	7.9977	7.9972
5.1.11	6.4523	7.9974	7.9973	7.9970	7.9971	7.9972
5.1.12	6.7057	7.9971	7.9971	7.9972	7.9972	7.9973
5.1.13	1.5483	7.9977	7.9975	7.9969	7.997	7.9971
5.1.14	7.3424	7.9974	7.9977	7.9972	7.997	7.9974
Average	6.0116	7.9973	7.9973	7.9972	7.9972	7.9973

5.7. NIST SP800-22 Test on Cipher Images

To further evaluate the statistical properties of the encrypted images, the NIST SP800-22 test suite is applied to the cipher images. Each encrypted image is converted into a binary sequence, and all 15 statistical tests are performed. The results are summarized in Table 6. As shown in Table 6, all *p*-values are greater than 0.01, indicating that the ciphertext sequences pass the randomness tests. For the template and excursions tests, the results are tabulated as pass ratios, which also demonstrate good randomness. Some results in the random excursions and random excursions variant tests are marked as "NA", since these tests require sufficient state cycles that may not be satisfied for sequences derived from images of limited size. Overall, the results confirm that the HESLM-IEA produces ciphertext sequences with strong randomness.

Table 6. NIST SP800-22 test results of the cipher images.

Tested Item	Tested Image											
	5.1.09		5.1.10		5.1.11		5.1.12		5.1.13		5.1.14	
Approximate Entropy	0.5478	P *	0.9746	P	0.0114	P	0.1755	P	0.8443	P	0.7201	P
Block Frequency	0.6352	P	0.7735	P	0.9024	P	0.4799	P	0.9653	P	0.6337	P
Cumulative Sums—forward	0.3137	P	0.4124	P	0.1353	P	0.6522	P	0.6117	P	0.9669	P
Cumulative Sums—reverse	0.2723	P	0.4670	P	0.2540	P	0.9734	P	0.4314	P	0.7633	P
FFT	0.4380	P	0.8117	P	0.8019	P	0.3874	P	0.3669	P	0.9677	P
Frequency	0.2415	P	0.2777	P	0.1648	P	0.6249	P	0.8273	P	0.7781	P
Linear Complexity	0.4551	P	0.1017	P	0.1023	P	0.0356	P	0.8154	P	0.7524	P
Longest Run	0.5603	P	0.2035	P	0.7891	P	0.8147	P	0.9328	P	0.3154	P
Non-overlapping Template **	145/148	– ****	148/148	–	144/148	–	145/148	–	147/148	-	147/148	–
Overlapping Template	0.6441	P	0.6021	P	0.1040	P	0.8801	P	0.8597	P	0.6717	P
Random Excursions **	8/8	–	NA ***	–	8/8	–	NA	–	NA	–	8/8	–
Random Excursions Variant **	18/18	–	NA	–	18/18	–	NA	–	NA	–	18/18	–
Rank	0.6379	P	0.3026	P	0.3089	P	0.8851	P	0.6879	P	0.7324	P
Runs	0.4547	P	0.0886	P	0.8446	P	0.6291	P	0.1724	P	0.2621	P
Serial Test P-value 1	0.7191	P	0.7968	P	0.4310	P	0.5864	P	0.2618	P	0.3865	P
Serial Test P-value 2	0.7786	P	0.7425	P	0.8678	P	0.6194	P	0.2775	P	0.4103	P
Universal	0.9641	P	0.7095	P	0.8465	P	0.2651	P	0.8167	P	0.9876	P

* "P" denotes that the sequence passes the test at the significance level of 0.01, while "F" indicates failure. ** For the non-overlapping template, random excursions, and random excursions variant tests, the results are reported as the number of subtests passed over the total number of tests. *** "NA" denotes that the test is not applicable due to insufficient state cycles or sequence conditions required by the NIST SP800-22 standard. **** "–" indicates that no separate pass or fail indicator is assigned for this test. The result is either expressed as a pass ratio or marked as "NA".

5.8. Differential Attack

Number of Pixel Change Rate (NPCR) and Unified Average Change Intensity (UACI) are two main metrics to evaluate the resistance against differential attacks, and their definitions measure the pixel intensity differences between two encrypted images corresponding to slightly different plaintexts, as given in Equations (27) and (28) [44]:

$$NPCR = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} D(i,j) = \begin{cases} 1 & \text{if } c_1(i,j) \neq c_2(i,j) \\ 0 & \text{if } c_1(i,j) = c_2(i,j) \end{cases}}{M \times N} \times 100\%, \quad (27)$$

$$UACI = \frac{1}{M \times N}\left(\sum_{i=1}^{M}\sum_{j=1}^{N} \frac{|c_1(i,j) - c_2(i,j)|}{255}\right) \times 100\%, \quad (28)$$

where M and N represent the row and column. For an IEA, the expected values of NPCR and UACI for 8-bit images are approximately 99.6094% and 33.4635% [45], respectively.

As tabulated in Table 7, the HESLM-IEA reaches or closely approaches these ideal values. Specifically, the NPCR values are consistently around 99.6086%, and the UACI values are approximately 33.4672%, which are very close to the theoretical expectations. In addition, compared with the existing IEAs, the HESLM-IEA achieves comparable or slightly better performance in most cases, indicating its strong diffusion capability. These results demonstrate that a slight change in the plaintext can lead to significant differences in the ciphertext, confirming that HESLM-IEA exhibits strong resistance against differential attacks.

Table 7. $NPCR(\%)$ and $UACI(\%)$ results.

Image	*NPCR*(%)				*UACI*(%)			
	Ref. [39]	**Ref. [42]**	**Ref. [43]**	**Our**	**Ref. [39]**	**Ref. [42]**	**Ref. [43]**	**Our**
5.1.09	99.6124	99.6337	99.6094	99.6185	33.4691	33.5125	33.4817	33.5038
5.1.10	99.6078	99.6002	99.6185	99.6445	33.4545	33.4745	33.4672	33.4829
5.1.11	99.6124	99.6032	99.6048	99.6033	33.4606	33.4245	33.4216	33.4597
5.1.12	99.6002	99.6078	99.6094	99.6078	33.4557	33.4847	33.4362	33.4390
5.1.13	99.617	99.617	99.6109	99.6017	33.4651	33.4365	33.4393	33.4528
5.1.14	99.6033	99.6078	99.6002	99.6124	33.4776	33.5190	33.4646	33.4647
Mean	99.6086	99.6116	99.6089	99.6147	33.4638	33.4753	33.4518	33.4672

5.9. Noise and Data Loss Attacks

The cipher image may be affected by noise and data loss during transmission and storage. Therefore, a robust IEA should withstand noise interference and partial data loss. In this section, the robustness of the HESLM-IEA is evaluated by introducing salt-and-pepper noise with increasing intensity (Figure 14) and applying localized data removal to the ciphertext (Figure 15). The results show that even under strong noise contamination, the decrypted images still preserve the main visual structures of the original image. In addition, the plaintext can be partially recovered even when up to 50% of the ciphertext is lost. These observations demonstrate that the HESLM-IEA maintains reliable decryption performance under noise and data loss, supporting its applicability in practical transmission environments.

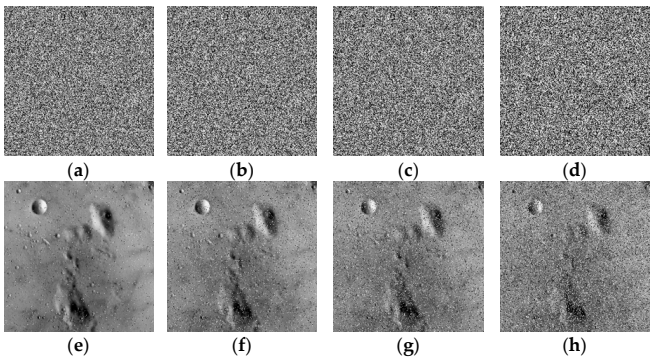
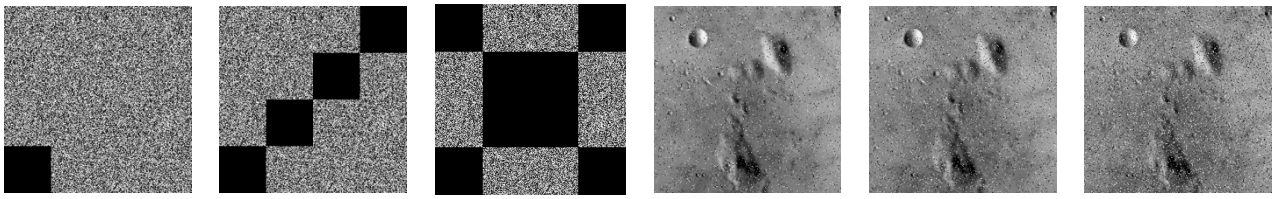


Figure 14. Noise attacks: (**a**) 0.01, (**b**) 0.05, (**c**) 0.1, and (**d**) 0.3 intensity on the cipher image "5.1.09"; (**e**–**h**) the decoded images.



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

Figure 15. Cropping attacks: (**a**) 1/16, (**b**) 1/4, and (**c**) 1/2 of the encrypted image "5.1.09"; (**d**–**f**) the decoded images.

5.10. Computational Complexity

To evaluate the practicality of the HESLM-IEA, its computational complexity was analyzed for an input image of size $M \times N$. The chaotic sequence generation requires $O(MN)$ operations, while the fractal-based scrambling introduces an additional logarithmic factor due to recursive partitioning and sorting, resulting in $O(MN \log(MN))$ complexity. The cascaded diffusion operates in $O(MN)$. Therefore, the overall time complexity is $O(MN \log(MN))$, which is dominated by the scrambling process. In terms of memory requirements, the HESLM-IEA mainly stores the original image, intermediate scrambled data, and a small number of auxiliary chaotic sequences, leading to a space complexity of $O(MN)$. No large-scale additional data structures are required, ensuring low memory overhead. Regarding scalability, the HESLM-IEA is well-suited for parallel implementation, as both the chaotic sequence generation and diffusion operations can be executed independently across pixels or blocks. This property makes the HESLM-IEA applicable to real-time and resource-constrained scenarios.

5.11. Comprehensive Comparison of Experimental Results

To provide a comprehensive evaluation of the HESLM-IEA, the average results of multiple test images are used in comparison with existing IEAs. As summarized in Table 8, all IEAs achieve SIE values close to 8 and NPCR/UACI values near the ideal levels, indicating strong randomness and resistance to differential attacks. For correlation, all IEAs produce values close to zero. Ref. [42] achieves the lowest correlation, while the proposed HESLM-IEA obtains slightly higher but still small values, indicating effective decorrelation. Overall, the HESLM-IEA demonstrates competitive or superior performance across multiple evaluation metrics. This improvement can be attributed to the enhanced chaotic dynamics of the 2D-HESLM and the effective integration of fractal-based scrambling and cascaded diffusion mechanisms.

Table 8. Performance comparison with existing methods (average results).

IEA	SIE	NPCR (%)	UACI (%)	Correlation		
				H	V	D
Ref. [39]	7.9973	99.6116	33.4638	0.0027	0.0038	0.0037
Ref. [42]	7.9972	99.6089	33.4753	0.0004	0.0002	0.0004
Ref. [43]	7.9972	99.6147	33.4672	0.0039	0.0024	0.0023
HESLM-IEA	7.9973	99.6147	33.4672	0.0020	0.0020	0.0017

6. Conclusions

This paper proposes a novel IEA based on a fractal–chaotic hybrid architecture. A 2D-HESLM is constructed to enhance chaotic behavior and enlarge the effective chaotic region. In addition, a fractal-inspired block scrambling strategy is designed to improve spatial permutation, and a cascaded diffusion mechanism is introduced to strengthen

pixel-level dependency. Extensive experimental results demonstrate that the HESLM-IEA achieves strong security performance. The information entropy of the encrypted images is close to 7.9980, while the NPCR and UACI values are approximately averaged to 99.6147% and 33.4672%, respectively, which are close to the theoretical expectations of an ideal IEA. The correlation coefficients of adjacent pixels are near zero in all directions, and both the chaotic sequences and ciphertext successfully pass the NIST SP800-22 statistical tests, confirming strong randomness properties. Despite these advantages, the HESLM-IEA still has some aspects that can be further improved. Computational complexity, mainly introduced by the cascaded diffusion and fractal-based operations, may increase processing time for large-scale images. However, this overhead projector remains manageable and can be optimized through more efficient implementation strategies. In addition, the recursive structure of the fractal-based scrambling may introduce scalability challenges when handling high-resolution images, as the number of partitioning operations increases with image size. Furthermore, the performance of the HESLM-IEA depends on the selection of initial parameters, which may influence the stability of the generated chaotic sequences. Future work will focus on addressing these aspects by reducing computational costs and improving efficiency, particularly for real-time and resource-constrained scenarios. In addition, the HESLM-IEA will be extended to more complex data modalities, such as video data and 3D medical imaging. Moreover, hardware-oriented implementations, including FPGA-based acceleration, will be explored to further enhance real-time performance and practical applicability.

The main abbreviations and symbols are summarized in Tables 9 and 10.

Table 9. List of abbreviations used in this paper.

Abbreviation	Full Name
IEA	Image encryption algorithm
CM	Chaotic map
2D-HESLM	Two-dimensional hyperbolic–exponential Sine–Logistic map
BD	Bifurcation diagram
LE	Lyapunov exponent
PE	Permutation entropy
HESLM-IEA, HESLM-IDA	HESLM-based image encryption and decryption algorithms
SC	Sierpiński carpet
NPCR	Number of pixel change rate
UACI	Unified average changing intensity
SIE	Shannon information entropy

Table 10. List of symbols and notations.

Symbol	Description	Symbol	Description
λ	LE value	X, Y	Chaotic sequences of the 2D-HESLM
s	Iteration level	CS_1–CS_6	Sub-chaotic sequences of X and Y
L_{node}	Length of the fractal graph	PI	Plain image
N_{node}	Number of nodes	HM	Hash-derived sequence
d_{H}	Hausdorff dimension	$offset$	Starting index for sub-sequence extraction
F_{n}	Fractal set at the n-th iteration	$h \times w$	Size of the current sub-block
F_{∞}	Limit fractal set	$ss_{i=1,2,3}$	Sub-sequences extracted for permutation
T_i	Affine transformation	img_{Vec}	One-dimensional representation of the image
K_{HV}	SHA-256 hash value	mix_{IDV}	Intermediate diffusion variable
K_{B}	Random bit sequence	W_{DWV}	Dynamic weight vector
bit_{i}	i-th bit of the K_{B}	V_{CLS}	Vector of the cyclic left shift
x_{init}	Computed value of K_{B}	b_{SBVI}	Shift bit value index
IT	Iteration times	SA	State vector
r, h, x_0, y_0	Initial parameters of the 2D-HESLM	EI	Encrypted image

Author Contributions: Conceptualization, Z.Z. and W.Z.; methodology, Z.Z. and N.R.; software, Z.Z. and W.Z.; validation, Z.Z., W.Z. and M.W.; formal analysis, Z.Z. and N.R.; investigation, Z.Z., P.Z. and Y.L.; resources, Z.Z.; data curation, Z.Z. and P.Z.; writing—original draft preparation, Z.Z.; writing—review and editing, Z.Z. and M.W.; visualization, Z.Z.; supervision, Z.Z. and Y.L.; project administration, W.Z.; funding acquisition, W.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the Research on Power Dispatching Technology for New Energy Storage Units Based- on Intelligent Monitoring System (No: LJ212411632076).

Data Availability Statement: No new data were created or analyzed in this study.

Conflicts of Interest: Author Zeyu Zhang is employed by Shenyang Huanke Technology Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as potential conflicts of interest.

References

1. Kocak, O.; Erkan, U.; Toktas, A.; Gao, S. PSO-based image encryption scheme using modular integrated logistic exponential map. *Expert Syst. Appl.* **2024**, *237*, 121452. [CrossRef]
2. Toktas, A.; Erkan, U.; Ustun, D.; Lai, Q. Multiobjective design of 2D hyperchaotic system using leader pareto grey wolf optimizer. *IEEE Trans. Syst. Man Cybern.* **2024**, *249*, 5237–5247. [CrossRef]
3. Alexan, W.; Hosny, K.; Gabr, M. A new fast multiple color image encryption algorithm. *Clust. Comput.* **2025**, *28*, 325. [CrossRef]
4. Al-Muhammed, M.; Ababneh, A.; Al-Daraiseh, A. Image encryption using color space neural network transformation and chaotic pixel perturbation. *Signal Image Video Process.* **2025**, *29*, 743. [CrossRef]
5. Benyahia, K.; Khobzaoui, A.; Benbakreti, S. Hybrid image encryption: Leveraging DNA sequencing and Lorenz chaotic dynamics for enhanced security. *Clust. Comput.* **2025**, *28*, 218. [CrossRef]
6. Lin, Y.; Xie, Z.; Chen, T.; Cheng, X.; Wen, H. Image privacy protection scheme based on high-quality reconstruction DCT compression and nonlinear dynamics. *Expert Syst. Appl.* **2024**, *257*, 124891. [CrossRef]
7. Zhao, C.; Zhai, Z.; Zeng, B. Delayed chaotic image encryption algorithm using cross-layer and DNA coding techniques. *Comput. Stand. Inter.* **2025**, *93*, 103974. [CrossRef]
8. Tang, M.; Du, G.; Lin, Y. A novel multi-color image compression encryption algorithm based on the reconstruction coefficient matrix and DNA point mutation operation. *Expert Syst. Appl.* **2025**, *270*, 126620. [CrossRef]
9. Huang, H.; Liu, Z.; Wang, Z.; Yan, F. A secure image encryption mechanism using biased Fourier quantum walk and addition-crossover structure in the Internet of Things. *Internet Things* **2025**, *31*, 101559. [CrossRef]
10. Liao, Y.; Lin, Y.; Xing, Z.; Li, Q.; Huang, G.; Chen, D.; Yuan, X. Using 3D-LMM-based encryption to secure digital images with 3-D S-box and Fibonacci Q-matrix. *IEEE Internet Things J.* **2025**, *12*, 55182–55195. [CrossRef]
11. Khadem, B.; Gholamzadeh, M.; Ahmad, M.; Ansari, Z.A. Robust image encryption scheme based on 6D hyper-chaos and DPA-resistant S-box. *Phys. Scr.* **2025**, *100*, 035213. [CrossRef]
12. Gao, S.; Iu, H.C.; Erkan, U.; Simsek, C.; Toktas, A.; Cao, Y.; Wu, R.; Mou, J.; Li, Q.; Wang, C. A 3D memristive cubic map with dual discrete memristors: Design, implementation, and application in image encryption. *IEEE Trans. Circuits Syst. Video Technol.* **2025**, *35*, 7706–7718. [CrossRef]
13. Lai, Q.; Yang, L.; Hu, G.; Guan, Z.; Iu, H.H.C. Constructing multiscroll memristive neural network with local activity memristor and application in image encryption. *IEEE Trans. Cybern.* **2024**, *54*, 4039–4048. [CrossRef]
14. Yang, F.; Ma, J.; Wu, F. Review on memristor application in neural circuit and network. *Chaos Solitons Fractals* **2024**, *187*, 115361. [CrossRef]
15. Yu, F.; Wu, Y.; Wang, X.; He, T.; Zhang, S.; Jin, J. New discrete memristive hyperchaotic map: Modeling, dynamic analysis, and application in image encryption. *Front. Phys.* **2025**, *13*, 1617964. [CrossRef]
16. Yu, F.; Xu, S.; Lin, Y.; Gracia, Y.M.; Yao, W.; Cai, S. Dynamic analysis, image encryption application and FPGA implementation of a discrete memristor-coupled neural network. *Int. J. Bifurcat. Chaos* **2024**, *34*, 2450068. [CrossRef]
17. Li, F.; Wang, Y.; Shen, H. Dual event-triggered synchronization of two-time-scale jumping neural networks and its application in image encryption and decryption. *IEEE Trans. Netw. Sci. Eng.* **2025**, *12*, 943–954. [CrossRef]
18. Lin, Y.; Liao, Y.; Zeng, W.; Wei, Y.; Chen, D.; Yuan, X. 3D non-degenerate hyperchaos: Design, analysis, and application in image encryption. *IEEE Trans. Consum. Electr.* **2026**, *Early Access*. [CrossRef]
19. Lin, Y.; Liao, Y.; Wei, Y.; Zeng, W.; Erkan, U.; Toktas, A.; Zhang, Y.; Chen, D. Lightweight image encryption via four-dimensional Hénon memristor map and fast block permutation. *Nonlinear Sci. Control Eng.* **2025**, *Epub ahead of print*. [CrossRef]

20. Ding, S.; Shi, F.; Erkan, U.; Toktas, A.; Li, Q.; Wang, C.; Gao, S.; Mou, J. Design of a three-dimensional logistic map and its application to seafood image encryption. *J. Supercomput.* **2026**, *82*, 225. [CrossRef]
21. Ding, S.; Wang, X.; Zhu, P.; Erkan, U.; Toktas, A.; Li, Q.; Mou, J.; Gao, S. SRA: A sine-based reconstruction approach to improve 2D chaotic map performance. *Eur. Phys. J. Plus* **2026**, *141*, 182. [CrossRef]
22. Lai, Q.; Liu, Y. A family of image encryption schemes based on hyperchaotic system and cellular automata neighborhood. *J. Sci. China Technol. Sci.* **2025**, *68*, 1320401. [CrossRef]
23. Tang, J.; Zhang, Z.; Huang, T. Two-dimensional cosine-sine interleaved chaotic system for secure communication. *IEEE Trans. Circuits Syst. II Exp. Briefs* **2024**, *71*, 2479–2483. [CrossRef]
24. Teng, L.; Cao, P.; Liu, Y. Multi-image encryption algorithm based on novel spatiotemporal chaotic system and dynamical chaotic trajectories. *IEEE Trans. Circuits Syst. Video Technol.* **2025**, *35*, 1562–1575. [CrossRef]
25. Li, C.; Li, Y.; Yu, W.; Moroz, I.; Volos, C. Constructing conditional symmetry in a chaotic map. *Nonlinear Dyn.* **2025**, *113*, 3857–3868. [CrossRef]
26. Yan, M.; Liu, M.; Li, C. DNA color image encryption based on conservative chaotic system. *Sci. Rep.* **2025**, *15*, 9278. [CrossRef] [PubMed]
27. Zhang, W.; Zhang, Z.; Wang, M.; Ren, N.; Wang, L.; Zhang, J.; Zhao, Y.; Wang, H. A new 2D-PECLSM map with hyperchaotic properties for image encryption application. *Int. J. Bifurcat. Chaos* **2025**, *35*, 2550048. [CrossRef]
28. Liu, R.; Wang, S.; Wang, Z.; Gong, X.; Zhang, S.; Zhou, J. A new color image encryption scheme based on DNA sequence operations and novel hyper-chaotic system. *Phys. Scr.* **2025**, *100*, 025025. [CrossRef]
29. Zhang, X.; Liu, M.; Niu, Y. Facial image encryption scheme based on improved 4-D hyperchaotic system. *J. Supercomput.* **2025**, *81*, 454. [CrossRef]
30. Wang, M.; Wang, X.; Zhao, T.; Zhang, C.; Xia, Z.; Yao, N. Spatiotemporal chaos in improved cross coupled map lattice and its application in a bit-level image encryption scheme. *Inf. Sci.* **2021**, *544*, 1–24. [CrossRef]
31. Sharma, M. Image encryption based on a new 2D logistic adjusted logistic map. *Multimed. Tools Appl.* **2020**, *79*, 355–374. [CrossRef]
32. Gao, X. Image encryption algorithm based on 2D hyperchaotic map. *Opt. Laser Technol.* **2021**, *142*, 107252. [CrossRef]
33. Teng, L.; Wang, X.; Xian, Y. Image encryption algorithm based on 2D-CLSS hyperchaotic map using simultaneous permutation and diffusion. *Inf. Sci.* **2022**, *605*, 71–85. [CrossRef]
34. Wang, M.; Fu, X.; Teng, L.; Yan, X.; Xia, Z.; Liu, P. A new 2D-HELS hyperchaotic map and its application on image encryption using RNA operation and dynamic confusion. *Chaos Solitons Fractals* **2024**, *183*, 114959. [CrossRef]
35. Qian, K.; Xiao, Y.; Wei, Y.; Liu, D.; Wang, Q.; Feng, W. A robust memristor-enhanced polynomial hyper-chaotic map and its multi-channel image encryption application. *Micromachines* **2023**, *14*, 2090. [CrossRef] [PubMed]
36. Pal, P.K.; Kumar, D. Zirili map-based image encryption method for healthcare, military, and personal data security. *Phys. Scr.* **2024**, *99*, 1125228. [CrossRef]
37. Pascual-Francisco, J.B.; Susarrey-Huerta, O.; Farfan-Cabrera, L.I.; Flores-Hernández, R. Creep properties of a viscoelastic 3D printed Sierpinski carpet-based fractal. *Fractal Fract.* **2023**, *7*, 568. [CrossRef]
38. Wang, M.; Teng, L.; Zhou, W.; Yan, X.; Xia, Z.; Zhou, S. A new 2D cross hyperchaotic Sine-modulation-Logistic map and its application in bit-level image encryption. *Expert. Syst. Appl.* **2025**, *261*, 125328. [CrossRef]
39. Xu, X.; Song, X.; Liu, S.; Zhou, N.; Wang, M. New 2D hyperchaotic Cubic-Tent map and improved 3D Hilbert diffusion for image encryption. *Appl. Intell.* **2025**, *55*, 590. [CrossRef]
40. Alexan, W.; Youssef, M.; Hussein, H.H.; Ahmed, K.K.; Hosny, K.M.; Fathy, A.; Mansour, M.B.M. A new multiple image encryption algorithm using hyperchaotic systems, SVD, and modified RC5. *Sci. Rep.* **2025**, *15*, 9775. [CrossRef]
41. Zhu, S.; Deng, X.; Zhang, W.; Zhu, C. Secure image encryption scheme based on a new robust chaotic map and strong S-box. *Math. Comput. Simul.* **2023**, *207*, 322–346. [CrossRef]
42. Lai, Q.; Liu, Y. A cross-channel color image encryption algorithm using two-dimensional hyperchaotic map. *Expert. Syst. Appl.* **2023**, *223*, 119923. [CrossRef]
43. Liang, Q.; Zhu, C. A new one-dimensional chaotic map for image encryption scheme based on random DNA coding. *Opt. Laser Technol.* **2023**, *160*, 109033. [CrossRef]
44. Hao, J.; Mou, J.; Xiong, L.; Zhang, Y.; Gao, X.; Sha, Y. A novel color image encryption algorithm based on the fractional order laser chaotic system and the DNA mutation principle. *Multimed. Tools Appl.* **2022**, *81*, 559–587. [CrossRef]
45. Wu, Y.; Noonan, J.P.; Agaian, S. NPCR and UACI randomness tests for image encryption. *Cyber J. Multidiscip. J. Sci. Technol. J. Sel. Areas Telecommun.* **2011**, *1*, 31–38.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.