entropy

MDPI

Article

3D-TCM-Driven Bit-Level Image Encryption via S-Box Feedback Algorithm

Jie Zhang [1,*], Wenjie Zhou [1], Mingxu Wang [2] and Yiting Lin [3,4]

[1] School of Electronics and Information Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China; wjzhou@lzjtu.edu.cn
[2] College of Information Engineering, Dalian Ocean University, Dalian 116023, China; 15041119328@163.com
[3] University of Electronic Science and Technology of China Zhongshan Institute, Zhongshan 528406, China; yitinglin@ieee.org
[4] University of Electronic Science and Technology of China, Chengdu 611731, China
* Correspondence: flying_zj@163.com

Abstract

Most existing low-dimensional chaotic maps suffer from a limited dynamical complexity and dynamic degradation, which restrict their effectiveness in image encryption. To address this issue, a novel three-dimensional chaotic map (3D-TCM) was constructed to improve dynamical complexity and stability, and its superiority was verified through a dynamical analysis. Based on these advantages, a plaintext-related image encryption scheme was designed by combining bit-level permutation and S-box-based diffusion. The experimental results show that the proposed scheme achieved high information entropy, a low pixel correlation, and desirable NPCR and UACI values, and successfully passed NIST SP800-22 statistical tests, demonstrating a strong resistance to differential attacks and overall robustness.

Keywords: chaos-based encryption; hyperchaotic system; S-box construction; secure image encryption

1. Introduction

In our era of information explosion, digital images have become a primary medium for communication and data exchange. However, transmitting image data over open and insecure networks exposes them to unauthorized access and potential attacks, making robust image encryption techniques essential [1]. With the rapid growth of multimedia applications and networked environments, the demand for efficient and secure image protection mechanisms has become increasingly critical.

Chaos-based image encryption algorithms (IEAs) may provide an effective solution due to the inherent properties of chaotic maps (CMs), such as sensitivity to initial conditions, pseudo-randomness, and nonlinear dynamics, which naturally satisfy Shannon's requirements of confusion and diffusion [2,3]. Low-dimensional chaotic maps (LDCMs), such as the Sinesine map, are widely adopted in IEAs due to their simple structure and low computational cost. Based on LDCMs, numerous IEAs have been proposed [4–7]. To further improve the performance, various enhanced schemes based on LDCMs and their variants have also been investigated, including optimization-based methods, hybrid chaotic structures, and permutation–diffusion frameworks [8–14]. However, recent studies have indicated that many LDCMs suffer from a limited dynamical complexity, a small key space, and dynamic degradation under finite precision, which may weaken their

Check for updates

Academic Editor: Boris Ryabko

Received: 6 April 2026
Revised: 5 May 2026
Accepted: 6 May 2026
Published: 8 May 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.

resistance to cryptanalysis. Such limitations restrict their ability to generate sufficiently complex chaotic sequences, thereby affecting the overall security performance of the corresponding encryption schemes. In contrast, high-dimensional chaotic maps (HDCMs) exhibit richer dynamical behaviors, broader parameter spaces, and higher unpredictability, making them more suitable for secure image encryption. Accordingly, recent research has increasingly focused on constructing high-dimensional or hybrid CMs to enhance the security performance.

Numerous HDCM-based IEAs have been developed. Lai et al. proposed a fast parallel multi-image encryption scheme based on a dual-memristor CM, achieving a high efficiency and strong resistance to statistical and differential attacks [15]. Kareem et al. introduced an algebraically enhanced three-dimensional (3D) CM combined with hash-based initialization, significantly improving the nonlinear dynamics and overall encryption robustness [16]. Aibai et al. developed an enhanced AES framework driven by a 2D HDCM, demonstrating that integrating chaos with classical cryptography can effectively strengthen security mechanisms [17]. Duan et al. designed a multi-image encryption scheme based on an HDCM and dual S-boxes, enhancing both the encryption capacity and the resistance to various attacks [18]. Deepankumar et al. employed a 3D CM for bit-plane-level permutation and diffusion, showing an improved key sensitivity, a reduced pixel correlation, and a strong robustness against multiple attack scenarios [19]. Kocak et al. proposed a 2D HDCM with superior dynamical properties, which significantly enhances the confusion and diffusion capabilities of the IEA [20]. Liu et al. further introduced a 2D non-degenerate CM-based method to batch-generate high-nonlinearity keyed S-boxes while eliminating fixed points and short cycles, thereby enhancing the resistance to cryptographic attacks [21]. Despite these advances, many existing HDCM-based IEAs still suffer from several limitations, such as an insufficient key sensitivity, limited resistance to differential attacks, and reliance on fixed or deterministic traversal structures. In addition, the interaction between chaotic dynamics and encryption operations is often not sufficiently tight, which limits the effective utilization of chaotic properties in practical encryption processes. Moreover, the use of static or linearly driven chaotic sequences derived from CMs without adaptive control may reduce randomness and degrade the security performance in practical implementations.

Therefore, the key challenge lies in jointly improving the dynamical quality of CMs and their effective integration into IEAs. Specifically, it remains difficult to design CMs with a sufficiently high complexity and stability while simultaneously constructing encryption frameworks that can fully exploit these properties to achieve a strong key sensitivity and diffusion capability. Motivated by this, this paper proposes a three-dimensional trigonometric coupled map (3D-TCM) with enhanced chaotic properties and stability and further constructs a plaintext-related IEA (TCM-IEA) based on it. By tightly coupling chaotic sequence generation with encryption operations, the proposed IEA enhances both the randomness and security performance. The main contributions are summarized as follows:

1. A 3D-TCM was constructed that exhibits a wide chaotic range, multiple positive Lyapunov exponents, and a strong dynamical complexity, providing a reliable foundation for secure image encryption.
2. A plaintext-related TCM-IEA was proposed by integrating bit-level conditional permutation and S-box driven feedback diffusion, where the key generation mechanism combines SHA-512 and CMs to enhance the randomness and key sensitivity.
3. Extensive experiments and security analyses indicated that the TCM-IEA exhibits a high performance in statistical properties, differential resistance, and ro-

bustness against noise and data loss attacks, confirming its effectiveness and practical applicability.

The remainder of the study is organized as follows: Section 2 constructs the 3D-TCM and analyzes its chaotic properties. Section 3 details the TCM-IEA framework, covering key generation, bit-level conditional permutation and S-box driven feedback diffusion and following a decryption process. Section 4 evaluates the security and performance of the TCM-IEA through several classic attacks. Section 5 provides a concluding summary.

2. Related Work

2.1. Construction of the 3D-TCM

The sine map is a classical one-dimensional CM, defined by [22]:

$$x_{n+1} = \mu \sin(\pi x_n)/4, \tag{1}$$

where μ is the control parameter. x_n and x_{n+1} are the state variables of the CM at iteration n and $n+1$.

Although the sine map has a simple structure, its chaotic behavior exists only within a limited parameter range, which restricts its applicability in IEAs. To improve this issue, various coupling strategies have been investigated for CMs. For example, Gao et al. extended the sine–logistic CM by incorporating an exponential function [23]. Wang et al. constructed a 3D CM through a combination of sine and linear functions [24]. Wang et al. further enhanced the 3D sine-based CM by incorporating fractional-order calculus, thereby achieving a wider chaotic range [25]. However, these CMs rely heavily on linear combinations or basic algebraic operations, which limit their nonlinear expressive capability.

To expand the chaotic range, the 3D-TCM is defined as follows:

$$\begin{cases} x_{n+1} = (a \cdot \sin(\pi x_n) + b \cdot \cos(\pi y_n)) \bmod 1 \\ y_{n+1} = (a \cdot \sin(\pi y_n) + b \cdot \cos(\pi z_n)) \bmod 1, \\ z_{n+1} = (a \cdot \sin(\pi z_n) + b \cdot \cos(\pi x_n)) \bmod 1 \end{cases} \tag{2}$$

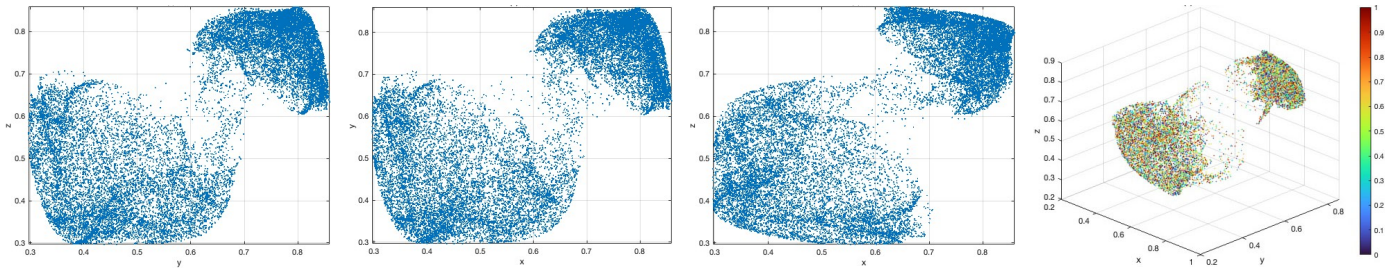
where a and b are control parameters. x_n, y_n, and $z_n \in [0,1)$ are the state variables of the 3D-TCM at iteration n, while x_{n+1}, y_{n+1}, and z_{n+1} represent the states at the next iteration. The symbol n denotes the iteration index. The mod denotes the modulo operation that maps the state variables into the interval $[0,1)$. To verify the chaotic properties of the 3D-TCM, we analyzed its dynamical behavior through a bifurcation diagram, phase trajectories, Lyapunov exponents, sample entropy, NIST tests, and a 0–1 test.

2.2. Attractor Analysis of the 3D-TCM

To further investigate the dynamical behavior of the 3D-TCM, this section analyzes its attractor behavior under different control parameters. By varying a and b and examining the evolution trajectories of the state variables x_n, y_n, and z_n, the geometric structures of the 3D-TCM in the 3D phase space can be clearly revealed.

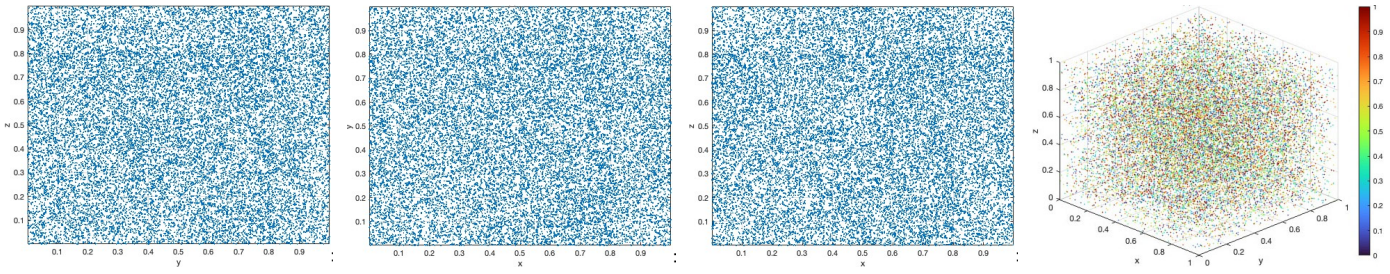
Figures 1 and 2 present the attractors corresponding to these parameter sets. For each case, the projections on the Y-Z, X-Y, X-Z, and X-Y-Z planes, as well as the 3D phase trajectories, are illustrated. From Figure 1, when $(a,b) = (0.8, 0.1)$, the 3D-TCM exhibits a typical structured chaotic attractor with a non-uniform density distribution, indicating strong nonlinear coupling among the state variables. As shown in Figure 2, when $(a,b) = (3,3)$, the trajectories are distributed almost uniformly in the state space, suggesting that the 3D-TCM approaches a higher level of chaotic complexity and mixing behavior. The results indicate that the 3D-TCM can obtain a wide variety of chaotic attractors with distinct

geometric structures. The 3D-TCM exhibits a strong sensitivity to parameter variations and initial conditions, as well as good phase space coverage and complexity.



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

Figure 1. Attractor of the 3D-TCM when $(a, b) = (0.8, 0.1)$. (**a**) y-z plane (**b**) x-y plane (**c**) x-z plane (**d**) x-y-z plane.

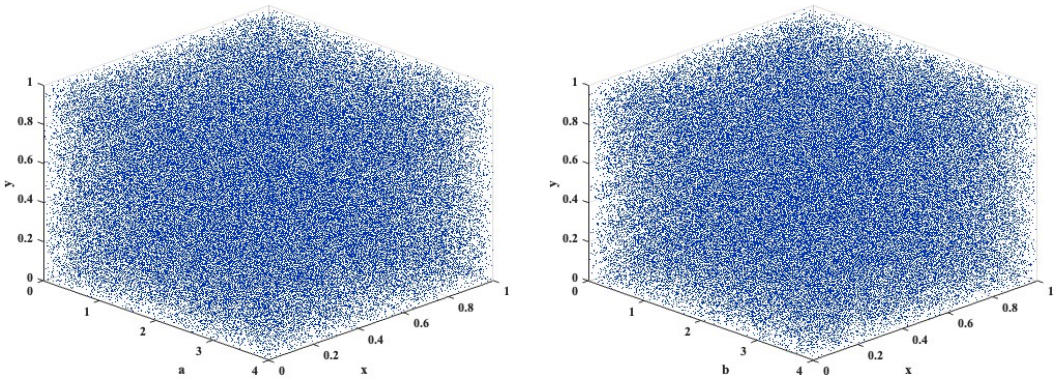


(**a**) (**b**) (**c**) (**d**)

Figure 2. Attractor of the 3D-TCM when $(a, b) = (3, 3)$. (**a**) y-z plane (**b**) x-y plane (**c**) x-z plane (**d**) x-y-z plane.

2.3. Bifurcation Diagram

A bifurcation diagram (BD) visualizes the dynamical behavior of a CM as the control parameter varies, revealing transitions from periodic motion to chaotic states [26]. As demonstrated in Figure 3, the 3D-TCM exhibits complex dynamical evolution over a wide parameter range. It is important to acknowledge that the BD reflects the CM's dynamical evolution rather than the statistical uniformity of trajectory distributions. Consequently, the nonuniform distribution of points in the BD is a common phenomenon in CMs and does not affect their chaotic properties.



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

Figure 3. BD of the 3D-TCM: (**a**) a and (**b**) b.

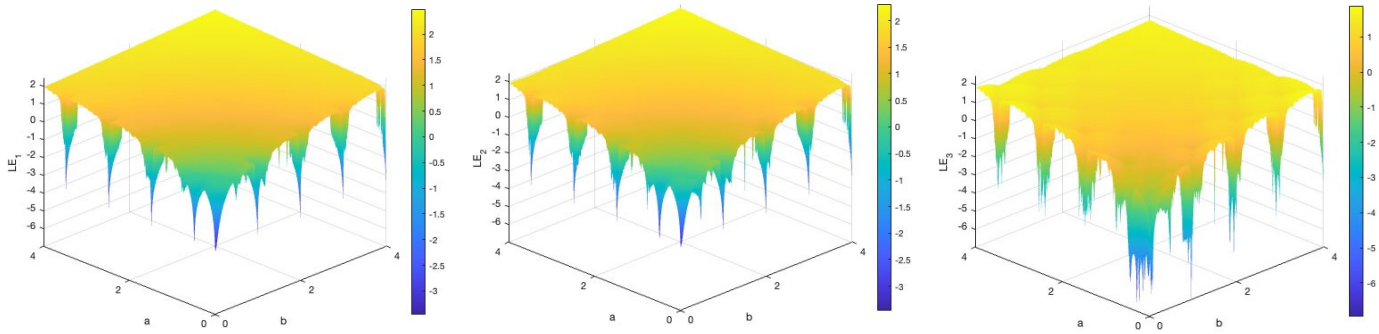
2.4. Lyapunov Exponent

The Lyapunov exponent (LE) quantifies the average exponential divergence rate of nearby trajectories in phase space and is widely used to characterize dynamical instability. In general, a positive LE indicates sensitive dependence on initial conditions, while the presence of multiple positive LEs suggests higher-dimensional complex dynamics, often associated with hyperchaotic characteristics. Furthermore, larger LE values generally correspond to stronger divergence behavior. The LEs can be calculated via [27]:

$$\lambda = \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln \frac{d_i}{\delta_0}, \tag{3}$$

where λ refers to the largest LE. n specifies the total number of iterations, i denotes the iteration index, d_i represents the Euclidean distance between two nearby trajectories at iteration i, and δ_0 is the initial perturbation magnitude.

As demonstrated in Figure 4, the 3D-TCM exhibits multiple positive LEs over a wide parameter range, indicating that the system possesses strong chaotic behavior and tends toward hyperchaotic dynamics in certain regions. In particular, the distributions of the three LEs over the (a, b) parameter plane reveal that the positive regions are not uniformly distributed, but appear in multiple disjoint intervals. This indicates that the 3D-TCM exhibits multi-regime dynamics with a strong sensitivity to initial conditions and complex divergence across multiple parameter sub-regions, enhancing the randomness and unpredictability in practical applications.



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

Figure 4. LE results of the 3D-TCM: (**a**) LE_1, (**b**) LE_2, and (**c**) LE_3.

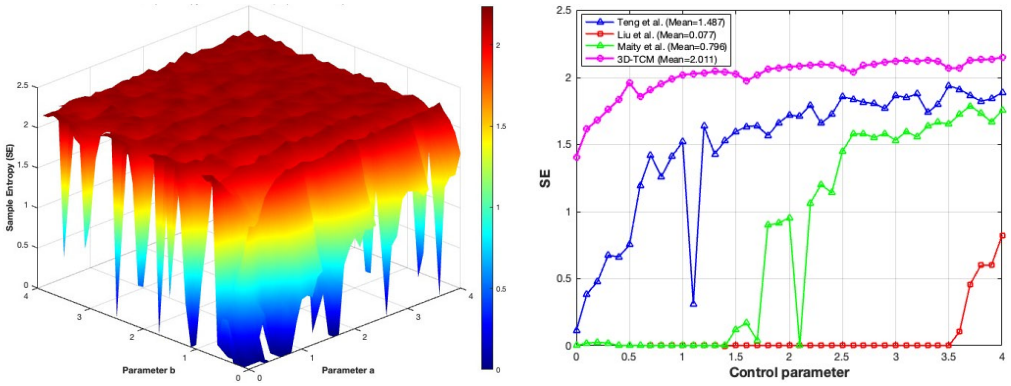
2.5. Sample Entropy

As posited by [28], sample entropy (SE) is a measure of the degree of disorder or unpredictability in a time series. A larger SE value indicates a higher randomness and a closer proximity to true randomness, which is a desirable property for key stream generation. It is imperative to note that, as the SE value of a CM increases, the difficulty of prediction also increases. In this section, the SE is computed by:

$$SE(m, r, N) = -\log \frac{S}{P}, \tag{4}$$

where m is the template dimension. r and N are the similarity threshold and the length of the time series, respectively. As demonstrated in Figure 5, the 3D-TCM exhibits consistently high SE values over a wide range of control parameters (a, b), with relatively small variation. The average SE value is significantly high, indicating that the 3D-TCM can generate sequences with a strong dynamical complexity. Furthermore, the regions with higher SE values are consistent with those where positive LEs are observed, suggesting that

the 3D-TCM possesses an excellent randomness generation capability. As further observed from Figure 5b, compared with conventional HDCMs [26,29,30], the 3D-TCM maintains higher SE values over a broader parameter range with smoother variation, indicating more stable chaotic dynamics and a stronger capability in generating pseudo-random sequences with an enhanced unpredictability and complexity.



(**a**) SE of the 3D-TCM (**b**) SE of the HDCMS

Figure 5. The SE of the 3D-TCM and the comparison with HDCMs [26,29,30].

2.6. NIST Test

The NIST SP800-22 comprises 15 statistical tests that are employed to detect non-randomness in binary sequences [31]. As demonstrated in Table 1, the results of all the tests indicate *p*-values that exceed the 0.01 significance threshold. This indicates that there is no rejection of the null hypothesis of randomness across all dimensions. It indicates that the sequences generated by the 3D-TCM exhibit a high degree of statistical randomness.

Table 1. NIST test results.

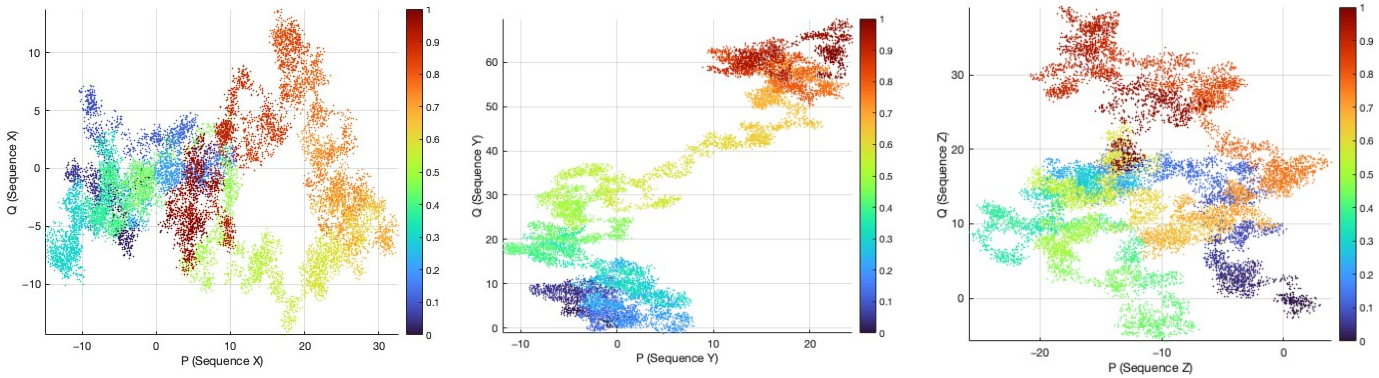
No.	Item	Sequence					
		x		*y*		*z*	
1	Approximate entropy	0.7545	*p* *	0.3071	*p*	0.2980	*p*
2	Block frequency	0.6899	*p*	0.5948	*p*	0.2804	*p*
3	Cumulative sums—forward	0.4891	*p*	0.1024	*p*	0.7329	*p*
	Cumulative sums—reverse	0.5095	*p*	0.0737	*p*	0.8156	*p*
4	FFT	0.8942	*p*	0.3636	*p*	0.2672	*p*
5	Frequency	0.3384	*p*	0.0981	*p*	0.9284	*p*
6	Linear complexity	0.2891	*p*	0.1458	*p*	0.0296	*p*
7	Longest run	0.6785	*p*	0.0537	*p*	0.7850	*p*
8	Non-overlapping template **	148/148	– ***	148/148	–	143/148	–
9	Overlapping template	0.1510	*p*	0.7605	*p*	0.0330	*p*
10	Random excursions **	8/8	–	8/8	–	7/8	–
11	Random excursion variant **	18/18	–	18/18	–	18/18	–
12	Rank	0.3101	*p*	0.9316	*p*	0.8059	*p*
13	Runs	0.3231	*p*	0.6583	*p*	0.0370	*p*
14	Serial test *p*-value 1	0.7978	*p*	0.2079	*p*	0.7437	*p*
	Serial test *p*-value 2	0.2612	*p*	0.2424	*p*	0.2990	*p*
15	Universal	0.4090	*p*	0.8926	*p*	0.9097	*p*

* "*p*" denotes that the sequence passes the test at the significance level of 0.01. ** For the non-overlapping template, random excursions, and random excursion 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.

2.7. Gottwald-Melbourne 0–1 Test

To assess the chaotic sequences produced by the 3D-TCM, the 0–1 chaos test introduced by Gottwald and Melbourne was employed. In this method, the time series is mapped to a

two-dimensional random walk. When the trajectory shows Brownian motion-like behavior, the sequence is considered random. The corresponding results are shown in Figure 6. It can be seen that the trajectories exhibit typical Brownian motion patterns, suggesting that the generated sequences have strong chaotic properties and a good randomness.



(**a**) Sequence X (**b**) Sequence Y (**c**) Sequence Z

Figure 6. Gottwald–Melbourne 0–1 test results when $(a, b) = (3.2, 2.7)$.

3. Encryption and Decryption Procedures

The employment of the 3D-TCM leverages its high sensitivity to initial conditions and control parameters, thereby ensuring the initialization of highly unpredictable pseudo-random sequences. This improves the resistance against attacks. The overall architecture of the TCM-IEA is depicted in Figure 7. The framework consists of three main modules: key generation, bit-level conditional permutation, and S-box-driven feedback diffusion. In the key stage, the plaintext image and the 3D-TCM are jointly utilized to generate the plaintext-related parameters and chaotic sequences required for encryption. Then, these sequences are used to control the permutation and diffusion processes. The remaining part of this section describes the specific TCM-IEA processes.

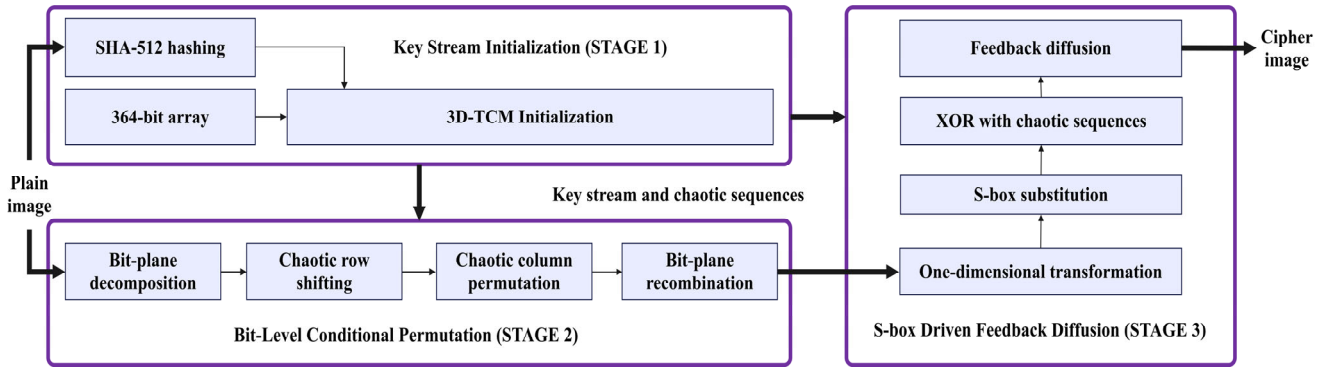


Figure 7. Block diagram of the TCM-IEA.

3.1. Generation of Keystream

For the plaintext image PI, a hash-based key generation mechanism is first applied to ensure plaintext sensitivity. The SHA-512 hash function is utilized to transform the image into a 128-character hexadecimal string K_{PI}, where each hexadecimal digit represents 4 bits. Then, 18-decimal values are calculated as follows:

$$HV_{\mathrm{PI}} = hex2dec(K_{\mathrm{PI}}), \tag{5}$$

where $hex2dec(\cdot)$ converts K_{PI} into the corresponding decimal values, which then undergo a modulus operation so that the resulting values lie within $[0, 255]$. Thereafter, two parameters FG and SG are calculated based on HV_{PI}:

$$\begin{cases} FG = round(mean(HV_{\mathrm{PI}}(1:9))) \\ SG = round(mean(HV_{\mathrm{PI}}(10:18))) \end{cases}, \tag{6}$$

where $mean(\cdot)$ represents the average value. $round(\cdot)$ denotes the rounding operation to the nearest integer. These values are then used to adaptively update the initial states and control parameters of the 3D-TCM. Given the preset parameters (a, b, x_0, y_0, z_0), the final parameters are obtained through a 364-bit random binary string K_{B} constructed following the IEEE 754 floating-point format, and this is then combined with the initial values and the control parameters using Algorithm 1 [32]:

$$x = \frac{\sum_{i=1}^{52} b_i 2^{52-i}}{2^{52}}, \tag{7}$$

where $b_i \in (0, 1)$ corresponds to the i-th bit of the binary key sequence K_{B}, and i represents the corresponding bit index. The 52-bit precision reflects the mantissa length of the IEEE 754 double-precision floating-point format, enabling the binary sequence to be converted into a real value within the interval $(0, 1)$.

Algorithm 1. Computational workflow of secret keys

Input: K_{B}, a, b, x_0, y_0, z_0, FG, SG
Output: a', b', x_0', y_0', z_0', $offset_{\mathrm{M}}$, $offset_{\mathrm{N}}$
$1: \ a' = a + (\sum_{i=1}^{52} K_{\mathrm{B}}[i] \times 2^{i-1}/2^{52}) \mathrm{mod}(4 - a)$
$2: \ b' = b + (\sum_{i=53}^{104} K_{\mathrm{B}}[i] \times 2^{i-53}/2^{52}) \mathrm{mod}(4 - b)$
$3: \ x_0' = x_0 + (\sum_{i=105}^{156} K_{\mathrm{B}}[i] \times 2^{i-105}/2^{52}) \mathrm{mod}(1 - x_0)$
$4: \ y_0' = y_0 + (\sum_{i=157}^{208} K_{\mathrm{B}}[i] \times 2^{i-157}/2^{52}) \mathrm{mod}(1 - y_0)$
$5: \ z_0' = z_0 + (\sum_{i=209}^{260} K_{\mathrm{B}}[i] \times 2^{i-209}/2^{52}) \mathrm{mod}(1 - z_0)$
$6: \ offset_{\mathrm{M}} = round(FG + (\sum_{i=261}^{312} K_{\mathrm{B}}[i] \times 2^{i-261}/2^{52}) \mathrm{mod}256)$
$7: \ offset_{\mathrm{N}} = round(SG + (\sum_{i=313}^{364} K_{\mathrm{B}}[i] \times 2^{i-313}/2^{52}) \mathrm{mod}256)$

Based on the updated parameters a', b', x_0', y_0', and z_0', three chaotic sequences X, Y, and Z and their corresponding integer forms X_{int}, Y_{int}, and Z_{int} are generated. The integer sequences are obtained through quantization:

$$\begin{cases} X_{\mathrm{int}} = floor(X \times 10^{15}) \mathrm{mod}256 \\ Y_{\mathrm{int}} = floor(Y \times 10^{15}) \mathrm{mod}256 \\ Z_{\mathrm{int}} = floor(Z \times 10^{15}) \mathrm{mod}256 \end{cases}, \tag{8}$$

where $floor(\cdot)$ denotes the floor operation that maps a real value to the largest integer not greater than it.

From the generated sequences, three key streams used in the diffusion stage are extracted according to the hash-dependent offsets HV_{PI}:

$$\begin{cases} CM_{\mathrm{diff}_1} = X_{\mathrm{int}}(1 + HV_{\mathrm{PI}}(2) : M \times N + HV_{\mathrm{PI}}(2)) \\ CM_{\mathrm{diff}_2} = Y_{\mathrm{int}}(1 + HV_{\mathrm{PI}}(3) : M \times N + HV_{\mathrm{PI}}(3)) \\ CM_{\mathrm{diff}_3} = Z_{\mathrm{int}}(1 + HV_{\mathrm{PI}}(4) : M \times N + HV_{\mathrm{PI}}(4)) \end{cases}, \tag{9}$$

where M and N denote the dimensions of the plaintext image PI.

This design ensures that the generated key stream is highly sensitive to the plaintext and exhibits a strong randomness.

3.2. TCM-IEA Process

The detailed procedure of TCM-IEA employs a permutation–diffusion framework that modifies both the positions and values of the pixels.

3.2.1. Bit-Level Conditional Permutation

Bit-level permutation starts with the original image PI. The process commences with the decomposition of PI into 8 bit-planes $BP = \{b_1, b_2, \ldots, b_8\}$. This is followed by organizing them into two bit-matrices, where the chaotic row-shifting operation and the chaotic column-reordering operation are performed using sequence X, Y, and Z along with the hash key HV_{PI}. The results of these operations are reshaped back into 8 bit-planes using chaotic indices and reordered before being fused together into the final scrambled image PI_{CON}. All permutation operations are performed on discrete bit arrays extracted from pixel values. In this context, the circular shift operator $cirshift(\cdot)$ denotes an index-based rotation of finite-length vectors rather than an operation on continuous chaotic signals. The detailed operation is presented in Algorithm 2.

Algorithm 2. Bit-level conditional permutation

Input: BP, X_{INT}, Y_{INT}, HV_{PI}
Output: PI_{CON}
$CM_1 \leftarrow reshape(X_{\mathrm{INT}}[HV_{\mathrm{PI}}(1)+1 : HV_{\mathrm{PI}}(1)+MN], M, N) \bmod 256$
$CM_{\mathrm{BP}(i)} \leftarrow bitget(CM_1, i)$, where $i = 1, 2, \ldots, 8$
$CM_2 \leftarrow reshape(X_{\mathrm{INT}}[HV_{\mathrm{PI}}(7)+1+MN : HV_{\mathrm{PI}}(7)+5MN], 2M, 2N)$
$CM_3 \leftarrow reshape(Y_{\mathrm{INT}}[HV_{\mathrm{PI}}(8)+1 : HV_{\mathrm{PI}}(8)+4MN], 2M, 2N)$
$idx_1 \leftarrow \arg_{\mathrm{sort}}(X_{\mathrm{INT}}[HV_{\mathrm{PI}}(1)+1 : HV_{\mathrm{PI}}(1)+8])$
$G_1 \leftarrow reshape([BP(idx_1(1:4))], 2, 2)$; $G_2 \leftarrow reshape([BP(idx_1(5:8))], 2, 2)$
$H_1 \leftarrow reshape([CM_{\mathrm{BP}}(idx_1(1:4))], 2, 2)$; $H_2 \leftarrow reshape([CM_{\mathrm{BP}}(idx_1(5:8))], 2, 2)$
$II \leftarrow X_{\mathrm{INT}}[5MN+1 : 5MN+2M]$; $LL \leftarrow X_{\mathrm{INT}}[5MN+2M+1 : 5MN+4M]$
for $i = 1$ to $2M$ do
    $s_1 \leftarrow II \bmod 2N$; $s_2 \leftarrow LL \bmod 2N$
    $dir_1 \leftarrow (-1)(\sum H_1(i,:) \bmod 2) \times (s_1 + HV_{\mathrm{PI}}(5))$
    $dir_2 \leftarrow (-1)(\sum H_2(i,:) \bmod 2) \times (s_2 + HV_{\mathrm{PI}}(6))$
    $K_1 \leftarrow circshift(G_1(i,:), dir_1)$; $K_2 \leftarrow circshift(G_2(i,:), dir_2)$
end for
for $j = 1$ to $2N$ do
    $ord_{CM_2} \leftarrow \arg_{\mathrm{sort}}(CM_2(:,j), \text{'descend'})$ if $\sum H_1(:,j)$ odd 'ascend'
    $ord_{CM_3} \leftarrow \arg_{\mathrm{sort}}(CM_3(:,j), \text{'descend'})$ if $\sum H_2(:,j)$ odd 'ascend'
    $O_1(:,j) \leftarrow K_1(idx_1(ord_{CM_2}), j)$; $O_2(:,j) \leftarrow K_2(idx_2(ord_{CM_3}), j)$
end for
$CI_{\mathrm{BP}}(k) \leftarrow subblock(O_1, k)$ for $k = 1, \ldots, 4$; $CI_{\mathrm{BP}}(k) \leftarrow subblock(O_2, k-4)$ for $k = 5, \ldots, 8$
$idx_3 \leftarrow \arg_{\mathrm{sort}}(Y_{\mathrm{INT}}[4MN+1 : 4MN+8])$
$CI'_{\mathrm{BP}} \leftarrow [CI_{\mathrm{BP}}(idx_3(1)), CI_{\mathrm{BP}}(idx_3(2)), \ldots, CI_{\mathrm{BP}}(idx_3(8))]$
$PI_{\mathrm{CON}} \leftarrow \sum_{i=1}^{8} CI_{\mathrm{BP}}{}'(i) \times 2^{i-1}$

3.2.2. S-Box-Driven Feedback Diffusion

An S-box without fixed points, reverse fixed points, or short-period cycles can be regarded as a strong S-box, as demonstrated in several existing studies [33–35]. In the TCM-IEA, a chaos-based S-box SB_{TCM} is constructed by utilizing the generated chaotic sequences X, Y, and Z. The SB_{TCM} is defined as a permutation of integers from 0 to 255. The chaotic sequences are then used to iteratively update SB_{TCM} through nonlinear transformations and swap operations. The following steps describe the generation process.

Step 1: The chaotic sequences X, Y, and Z are generated and reshaped into one-dimensional vectors X_{c}, Y_{c}, and Z_{c}. Meanwhile, SB_{TCM} is constructed as $SB_{\mathrm{TCM}} = \{0, 1, 2, \ldots, 255\}$, and a time-varying control parameter is introduced to enhance the dynamic behavior of the system.

Step 2: For each element, control parameters are derived from the chaotic sequences, and a nonlinear update mechanism is applied to produce a new state. The updated state is then mapped to an integer index within $[1, 256]$ through scaling and modulo operations.

Step 3: A swap operation is performed between the current position and the generated index in SB_{TCM}. Meanwhile, the chaotic states are updated through coupling relationships and normalized into the interval $[0, 1)$, ensuring continuous evolution of the system.

The above process is iteratively executed for a predefined number of iterations, resulting in a highly nonlinear and complex S-box with an enhanced randomness. The detailed operation is presented in Algorithm 3.

Algorithm 3. S-box-driven feedback diffusion

Input: X, Y, Z, T, L
Output: SB_{TCM}
for $t = 1$ **to** T **do**
  Update control parameter a and compute mean value of X
  for $i = 1$ **to** 256 **do**
    $X(i) = X_{\mathrm{c}}(i),\ Y(i) = Y_{\mathrm{c}}(i),\ Z(i) = Z_{\mathrm{c}}(i)$
    $r = Y(i),\ A = 2 \times a \times r - a,\ C = 2 \times r,\ p = Z(i)$
    if $p < 0.5$
      if $|A| < 1$
        $X_{\mathrm{new}} = mean(X) - A \times |C \times mean(X) - X(i)|$
      else
        $k = (Z(i) \bmod L) + 1$
        $X_{\mathrm{new}} = X(k) - A \times |C \times X(k) - X(i)|$
      end if
    else
      $l_i = -1 + 2Y(i)|$
      $X_{\mathrm{new}} = \left|mean(X) - X(i)\right| \times e^{l_i} \times \cos(\pi l_i) + mean(X)$
    end if
    $idx_i = (floor(\left|X_{new}\right| \times 10^6 + Y(i) \times 10^6 + Z(i) \times 10^6) \bmod 256) + 1$
    Swap $SB_{\mathrm{TCM}}(i)$ and $SB_{\mathrm{TCM}}(idx)$
    Update and normalize $X(i),\ Y(i),\ Z(i)$
  end for
end for
Reshape SB_{TCM} into 16×16 S-box
return SB_{TCM}

To further construct an S-box with a high nonlinearity while eliminating structural weaknesses, an adjustment strategy was employed. Starting from the resulting SB_{TCM}, pairs of elements are randomly exchanged, and only those swaps that improve nonlinearity are retained through iterative updates. Subsequently, fixed points and reverse fixed points are removed via local adjustments. Finally, disjoint cycles are connected into a single long cycle of a 256 length to avoid short periodic structures. The final SB_{TCM} is tabulated in Table 2. To evaluate its cryptographic performance, several widely used criteria were adopted, including nonlinearity, the bit independence criterion (BIC), the strict avalanche criterion (SAC), and the differential approximation probability (DAP) [36]. The corresponding experimental results are tabulated in Table 3, demonstrating that the S-box achieved a competitive performance.

Table 2. Constructed results of the S-box.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	B8	8F	4E	C1	44	CC	38	DA	BA	E7	1B	F7	08	77	95	49
1	98	70	9C	26	A0	A7	A5	61	AB	5B	B1	B7	D6	4B	C3	41
2	CE	36	52	29	06	DC	F6	56	19	79	F0	76	0E	71	9A	93
3	2E	68	11	2A	A9	85	AF	96	12	4F	BE	46	C8	3C	D2	D7
4	DE	24	EB	16	FA	05	5F	AD	58	B4	9B	BD	47	89	51	D3
5	30	DF	23	EC	15	FB	6E	69	62	F2	AA	5A	B3	3D	BC	45
6	C6	3B	D1	2F	DD	25	21	17	F9	07	7B	90	78	9D	74	50
7	6C	A1	64	67	D9	B0	72	B9	20	C4	3F	CF	83	D8	28	E4
8	1E	F1	10	0F	01	88	F5	09	81	80	5C	0A	7D	8E	7A	1A
9	92	99	E9	9F	1D	A6	BF	E0	55	6D	4A	C0	42	CA	39	D4
A	7C	73	27	E3	1F	EF	B6	FC	FF	5E	AC	53	BB	48	C5	3E
B	CD	33	91	34	E2	0D	ED	13	F4	0B	31	6F	97	66	A2	57
C	B5	4C	40	CB	35	2B	2C	E1	63	E6	54	A4	DB	FE	18	A3
D	FD	02	14	F8	04	2D	EE	65	8B	03	82	94	32	B2	75	59
E	60	C9	43	D5	37	6B	9E	E5	22	F3	0C	7F	C2	6A	D0	3A
F	1C	AE	E8	5D	8C	84	8D	EA	A8	4D	8A	86	0	87	C7	7E

Table 3. Comparative performance of the TCM-IEA and existing IEAs.

S-Box	Nonlinearity			BIC	SAC	DAP
	Min	Max	Mean			
Ours	104	108	106.25	102.79	0.5078	0.0391
AES	112	112	112	112	0.5048	0.0156
SM4	112	112	112	112	0.4997	0.0156
Ref. [37]	100	108	105	103	0.5002	0.0469
Ref. [38]	100	108	104	103.5	0.516	0.039
Ref. [39]	98	106	103.5	103.93	0.4958	0.313

To further enhance security, a further diffusion operation was introduced to propagate the influence of each pixel over the entire ciphertext.

Step 1: The plaintext image is reshaped into a one-dimensional sequence PI_{vec}, and three chaotic sequences X, Y, and Z are quantized and truncated to match the length of PI_{vec}.

Step 2: Each element of PI_{vec} is substituted using the constructed SB_{TCM} to obtain an intermediate sequence:

$$CI_3(i) = SB_{\mathrm{TCM}}(PI_{\mathrm{vec}}(i)). \tag{10}$$

Step 3: An initial diffusion value is generated by combining the substituted value and chaotic sequences:

$$CI(1) = CI_3(1) \oplus X(1) \oplus Y(1) \oplus Z(1). \tag{11}$$

Step 4: A cascading diffusion process is applied, where each ciphertext element depends on the previous one and the chaotic sequences:

$$CI(i) = CI_3(i) \oplus CI(i-1) \oplus X(i) \oplus Y(i) \oplus Z(i), \tag{12}$$

where $i \geq 2$.

3.3. Decryption Process

The decryption process, termed the 3D-TCM-based image decryption algorithm (TCM-IDA), is designed as the exact inverse of the proposed TCM-IEA. The original plaintext image can be accurately recovered from the ciphertext using the same secret keys and chaotic sequences.

Step 1: The CI is reshaped into a one-dimensional sequence, and the chaotic sequences X, Y, and Z are regenerated and quantized in the same manner as in the TCM-IEA.

Step 2: The cascading diffusion process is reversed. The intermediate sequence is obtained as:

$$CI_3 = CI(i) \oplus CI(i-1) \oplus X(i) \oplus Y(i) \oplus Z(i), \tag{13}$$

and for the first element,

$$CI_3(1) = CI(1) \oplus X(1) \oplus Y(1) \oplus Z(1). \tag{14}$$

Step 3: The initial diffusion is reversed by recovering, which directly corresponds to the substituted sequence before cascading diffusion.

Step 4: The inverse S-box substitution is performed using the inverse mapping of SB_{TCM}, which is a bijection over $[0, 255]$, to obtain:

$$PI_{\mathrm{vec}}(i) = SB_{\mathrm{TCM}}(CI_3(i)). \tag{15}$$

Step 5: The sequence PI_{vec} is reshaped into a two-dimensional image, which corresponds to the permuted image obtained after the bit-level conditional permutation in the encryption stage.

Step 6: The inverse bit-level conditional permutation is executed. All operations in Algorithm 2 are reversed in inverse order using the same chaotic index sequences idx_1, idx_2, and idx_3. Specifically, the inverse circular shift is applied with opposite directions, and the inverse sorting operations restore the original positions of rows, columns, and sub-blocks. Finally, the bit-planes are recommended to reconstruct the original plaintext image PI.

Since each component of the encryption process, including XOR-based diffusion, bijective S-box substitution, and index-driven permutation, is strictly invertible, the proposed scheme ensures exact and lossless recovery of the plaintext when the correct key is used.

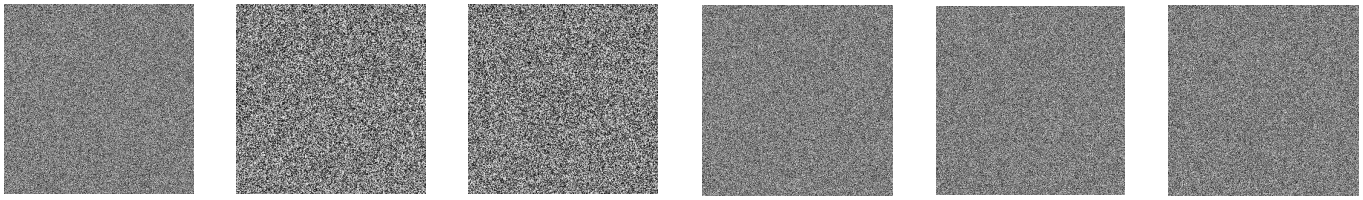
4. Experimental Results and Secure Analysis

In this section, a variety of standard grayscale images, primarily selected from the SIPI dataset, are employed to evaluate the performance of the TCM-IEA and TCM-IDA (Figure 8). Comprehensive experimental results are presented, including the histogram distribution, a correlation analysis and the information entropy. In addition, several typical attack scenarios are simulated on the ciphertext images to assess the robustness of the IEA.

The results demonstrate that the encrypted images can be correctly recovered, confirming the validity of the decryption process.



(**a**) Original images



(**b**) Cipher images



(**c**) Decoded images

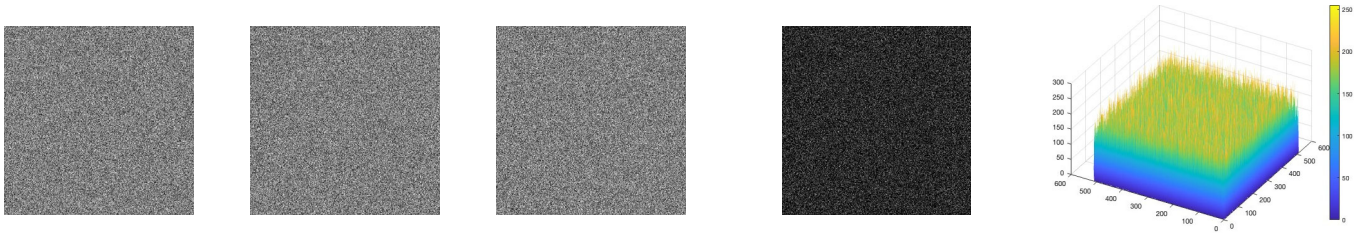
Figure 8. Experimental results of the images.

4.1. Key Space

The secret key of the TCM-IEA consists of three independent components. (1) The initial states and control parameters of the 3D-TCM are represented in double-precision format, with an effective precision of approximately 10^{-15}. Each parameter contributes about 2^{52} possible values. Considering the parameters (a, b, x_0, y_0, z_0), the corresponding key space is approximately $(2^{52})^5 = 2^{260}$. (2) A hash-based key HV_{PI} is generated from the plaintext image using the SHA-512 algorithm, providing a key space of 2^{512}, which introduces strong plaintext dependence. (3) An independent 364-bit random binary sequence is employed, contributing an additional key space of 2^{364} and further enhancing the randomness. Since these components are independent, the total key space of the 3D-TCM is sufficiently large to withstand brute-force attacks.

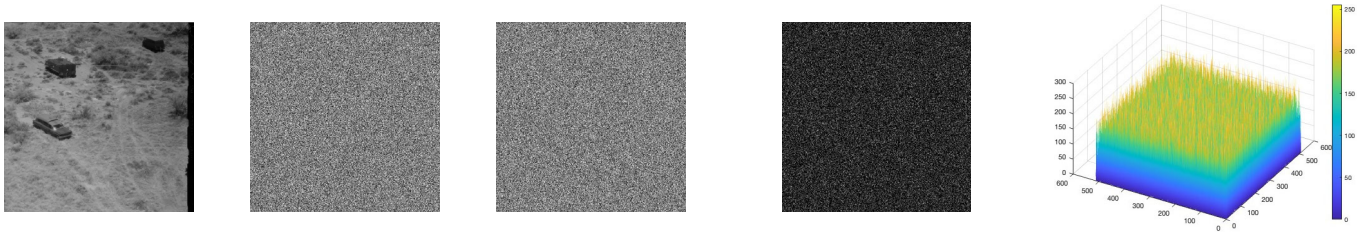
4.2. Key Sensitivity

The key sensitivity of the IEA was also evaluated by slightly perturbing the encryption keys [40], altering one bit of either the control parameter a or the key component HV_{PI} and then executing the encryption and decryption operation. As shown in Figure 9, even tiny changes in the key led to completely different ciphertexts, indicating that the encryption output is highly sensitive to initial key variations. From the decryption perspective, the sensitivity was further verified by attempting to recover the plaintext using slightly altered keys. As illustrated in Figure 10, only the exact original key can correctly reconstruct the plaintext, while any deviation results in meaningless noisy images. This demonstrates that the TCM-IDA is strictly dependent on the correct key and confirms that it serves as the exact inverse of the encryption operation, ensuring the practical realizability and high key sensitivity of the TCM-IDA.



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

Figure 9. Key sensitivity: the cipher image using (**a**) the original keys, (**b**) $a + 10^{-15}$, and (**c**) $HV_{\mathrm{PI}}(2) + 1$, (**d**) (**b**–**c**), (**e**) the histogram of (**d**).



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

Figure 10. Key sensitivity: the decoded image using (**a**) the original keys, (**b**) $a + 10^{-15}$, and (**c**) $HV_{\mathrm{PI}}(2) + 1$, (**d**) (**b**–**c**), and (**e**) the histogram of (**d**).

4.3. Statistical Analyses

4.3.1. Histogram

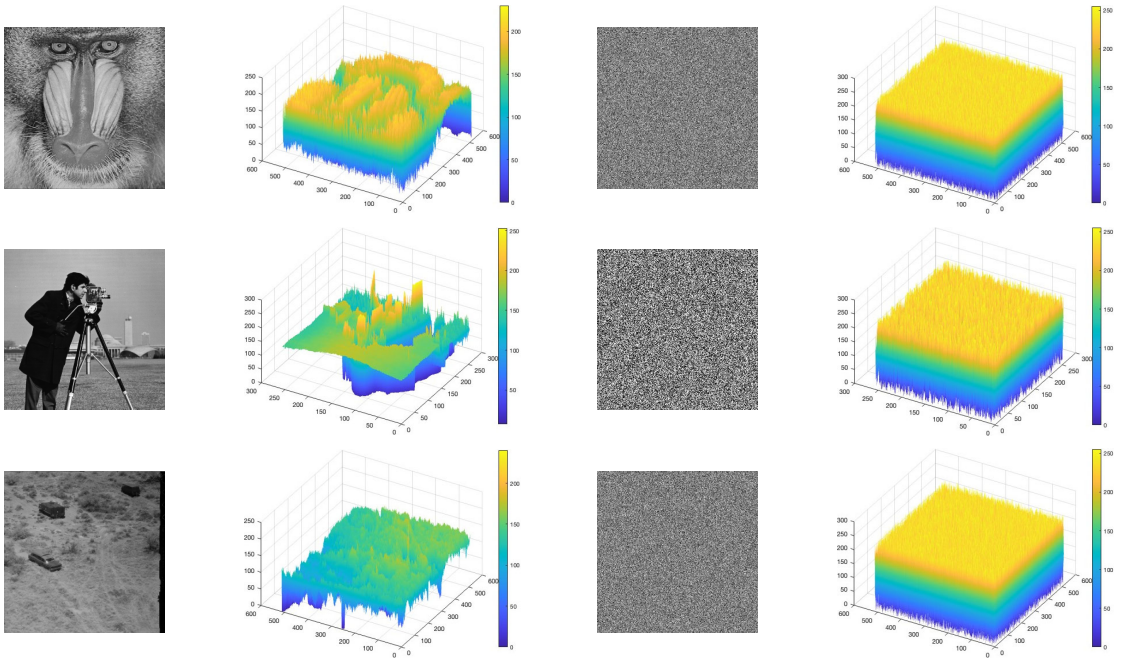
A histogram depicts the distribution of pixel intensities in an image and is commonly used to measure its statistical randomness and ability to withstand statistical attacks [41]. As depicted in Figure 11, the original images exhibited non-uniform histograms with prominent peaks, indicating redundant patterns. Their encrypted counterparts instead exhibited a uniform distribution across intensity levels. This demonstrates that the TCM-IEA diffuses the statistical information of the plaintext image, thereby enhancing the resistance to histogram-based statistical attacks.

4.3.2. Correlation

This section evaluates the capability of the IEA by analyzing the correlation among neighboring pixels in three different directions: horizontally (H), vertically (V) and diagonally (D). The correlation coefficient is calculated by [42]:

$$r_{x,y} = \frac{\sum_{i=1}^{t} (x_i - E(x))(y_i - E(y))}{\sqrt{\sum_{i=1}^{t} (x_i - E(x))^2} \sqrt{\sum_{i=1}^{t} (y_i - E(y))^2}}, \tag{16}$$

where t is the number of pairs of adjacent pixels in the image. $E(x)$ and $E(y)$ are their respective average values. As shown in Table 4 and Figure 12, the plaintext exhibited strong pixel correlation, which the blue, red, and yellow colors represent the three directions. As further observed from Table 4, the TCM-IEA generally achieved correlation coefficients closer to zero than those of Ref. [43], indicating a stronger decorrelation capability across the horizontal, vertical, and diagonal directions. The TCM-IEA can effectively break the pixel correlation.

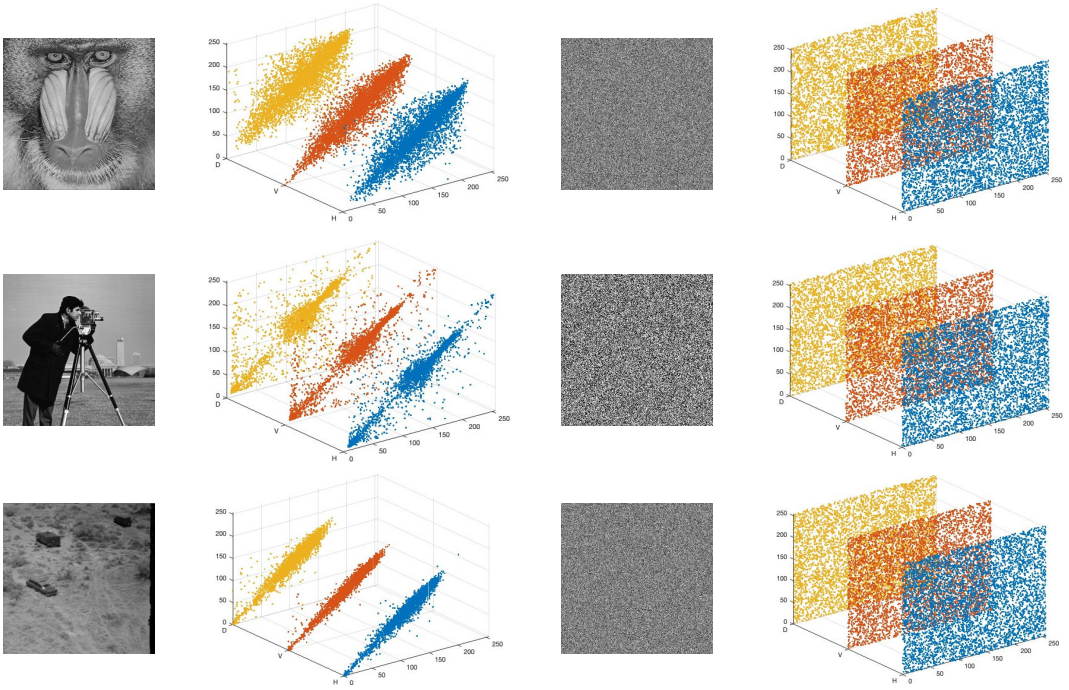


(**a**) Original images (**b**) Histograms of (**a**) (**c**) Cipher images (**d**) Histograms of (**c**)

Figure 11. Histograms of the original images and the encrypted images.

Table 4. Correlation coefficient results.

Test Image	Ours						Ref. [43]		
	Plain Image			Cipher Image			Cipher Image		
	H	V	D	H	V	D	H	V	D
Baboon	0.8665	0.7586	0.7261	−0.0012	−0.0002	0.0009	0.0018	-8.3384×10^{-4}	−0.0012
Cameraman	0.9335	0.9592	0.9087	0.0014	0.0027	0.0043	0.0023	−0.0014	−0.0039
Peppers	0.9659	0.9636	0.9325	−0.0013	0.0006	−0.0004	0.0026	0.0046	−0.0046



(**a**) Plain images (**b**) Correlation of (**a**) (**c**) Cipher images (**d**) Correlation of (**c**)

Figure 12. Three-directional correlation of the plain images and their corresponding cipher images.

4.3.3. Information Entropy

Information entropy is a measure of uncertainty for a data source [44]. In IEAs, it determines the randomness in the ciphertext of the image, and thus indicates how secure it is against statistical attacks. The largest entropy attainable for an 8-bit grayscale image that has 256 intensity values is 8, implying a uniform distribution of these 256 values, which gives the greatest level of randomness. It is calculated as follows [45]:

$$H(x) = -\sum_{i=1}^{n} P(x_i) \log_2(P(x_i)), \tag{17}$$

where n is the total number of pixels. $P(x_i)$ is the probability of occurrence of gray level x_i.

Table 5 tabulates the entropy values of the encrypted images, which were consistently close to the ideal value of 8. As can be observed from Table 5, the TCM-IEA achieved entropy values that were comparable to, and in some cases, slightly closer to the ideal value than, those of Refs. [32,43], indicating a more uniform distribution of pixel intensities in the ciphertext. Compared with existing IEAs, the TCM-IEA achieved a competitive performance, confirming its effectiveness against entropy-based attacks.

Table 5. Entropy values.

Image	IES		
	Ref. [32]	**Ref. [43]**	**Ours**
7.1.01	7.9993	7.9994	7.9993
7.1.02	7.9991	7.9993	7.9992
7.1.03	7.9993	7.9993	7.9994
7.1.04	7.9994	7.9993	7.9993
7.1.05	7.9992	7.9994	7.9992
7.1.06	7.9993	7.9993	7.9993

4.4. Differential Attack

Differential attacks assess a cipher's sensitivity to variations in the plaintext or key inputs, determining if an IEA has a stable avalanche effect. A good IEA produces different ciphertexts no matter how a single pixel is changed in a single image input. The two main metrics that are employed to measure the avalanche effect, named the number of pixel change rate (NPCR) and the unified average changing intensity (UACI) [46], are calculated as follows:

$$NPCR = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} D(i,j) = \begin{cases} 1 & \text{if } c_1(i,j) \neq c_2(i,j) \\ 0 & \text{if } c_1(i,j) = c_2(i,j) \end{cases}}{M \times N} \times 100\%, \tag{18}$$

$$UACI = \frac{1}{M \times N}\left(\sum_{i=1}^{M}\sum_{j=1}^{N} \frac{|c_1(i,j) - c_2(i,j)|}{255}\right) \times 100\%, \tag{19}$$

where $c_1(i,j)$ and $c_2(i,j)$ are the grayscale intensities at position (i,j). The NPCR measures the proportion of differing pixels between two images encrypted from nearly identical plaintexts, while the UACI quantifies the average intensity difference in those changed pixels. Under uniform randomness, the NPCR and UACI values for grayscale images are approximately 99.6094% and 33.4635%, respectively.

As shown in Table 6, the computed NPCR and UACI scores are close to the ideal values. As further observed from Table 6, the proposed method achieved NPCR and UACI values that were comparable to, and in some cases, slightly better than, those of Refs. [32,43], indicating a strong capability to detect and amplify small changes in the plaintext. This

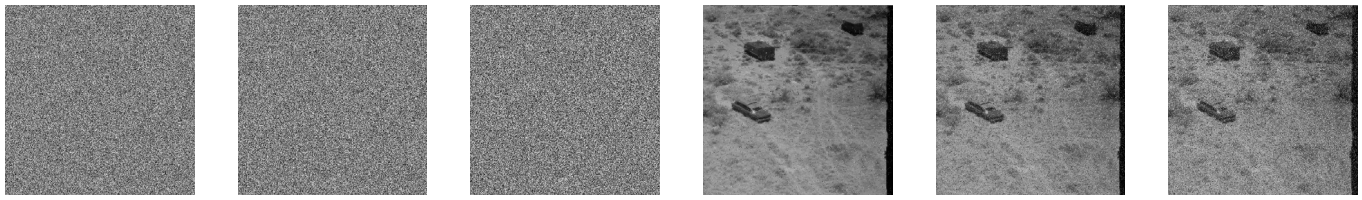
is mainly attributed to the strong diffusion capability of the TCM-IEA, which enables small variations in the plaintext to rapidly propagate across the entire ciphertext image, resulting in near-uniform statistical distributions. The results demonstrate that the TCM-IEA achieves a strong resistance against differential attacks. In addition, the standard deviations of NPCR and UACI across different test images were small, which further confirms the stability and statistical reliability.

Table 6. $NPCR(\%)$ and $UACI(\%)$ values.

Image	NPCR			UACI		
	Ref. [32]	Ref. [43]	Ours	Ref. [32]	Ref. [43]	Ours
7.1.01	99.6040	99.6262	99.6193	33.4694	33.4223	33.4834
7.1.02	99.6212	99.6147	99.6162	33.4410	33.5315	33.4065
7.1.03	99.6151	99.6193	99.6006	33.4801	33.4990	33.4192
7.1.04	99.6014	99.6025	99.6044	33.4845	33.4838	33.4994
7.1.05	99.6109	99.5956	99.6052	33.4939	33.4713	33.4661
7.1.06	99.6124	99.5983	99.6189	33.4338	33.4577	33.5128
Mean	99.6140	99.6147	99.6108	33.4663	33.4615	33.4646

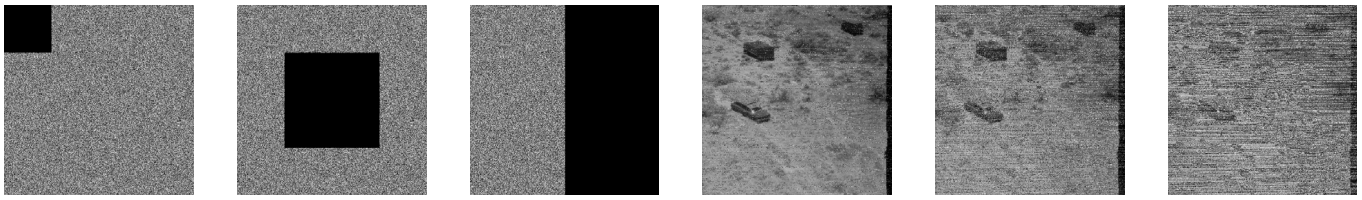
4.5. Noise and Data Loss Attacks

For an IEA, their cipher images should remain robust against the distortion from transmission or storage [47]. This section tests the TCM-IEA against two main indicators: salt-and-pepper noise at different densities and cropped attacks at different ratios. The restored images presented in Figures 13 and 14 still preserve the intact structural information and visual details, even though they have undergone significant distortions due to the introduction of large amounts of random noises and missing small blocks.



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

Figure 13. Noise resistance: (**a**) 0.01-, (**b**) 0.05-, and (**c**) 0.1-intensity salt and pepper of the encrypted image 5.1.09; (**d**–**f**) are the decoded images of (**a**–**c**).



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

Figure 14. Data loss: (**a**) 1/16, (**b**) 1/4, and (**c**) 1/2 of the encrypted image 5.1.09; (**d**–**f**) are the decoded images of (**a**–**c**).

4.6. NIST Test for Encrypted Images

To further validate the randomness of the TCM-IEA, the NIST SP800-22 test suite was also applied to the ciphertext sequences. Specifically, the encrypted images were first converted into one-dimensional binary sequences and then evaluated using the same 15 statistical tests as described in Section 2.6. As shown in Table 7, the p-values obtained

from all the tests were greater than the significance level of 0.01, indicating that the null hypothesis of randomness cannot be rejected. In addition, for tests such as the non-overlapping template, random excursions, and random excursions variant, the pass ratios satisfied the required criteria, demonstrating stable statistical behavior. Compared with the results of chaotic sequences, the ciphertext sequences also exhibited consistent randomness across all the tested dimensions. This indicates that the TCM-IEA not only preserves the randomness of the underlying 3D-TCM, but also effectively propagates it through the permutation–diffusion process. Therefore, the NIST test results on the encrypted images further confirm that the TCM-IEA produces ciphertexts with a high statistical randomness, providing strong support for its security and resistance.

Table 7. NIST test results of the cipher images.

No.	Item	Cipher Image					
		7.1.02		7.1.03		7.1.04	
1	Approximate entropy	0.6495	*p*	0.5052	*p*	0.4894	*p* *
2	Block frequency	0.9344	*p*	0.6135	*p*	0.4107	*p*
3	Cumulative sums—forward	0.8381	*p*	0.2279	*p*	0.665	*p*
	Cumulative sums—reverse	0.5882	*p*	0.1544	*p*	0.4046	*p*
4	FFT	0.3204	*p*	0.6907	*p*	0.4693	*p*
5	Frequency	0.5243	*p*	0.1362	*p*	0.7539	*p*
6	Linear complexity	0.7722	*p*	0.9718	*p*	0.0963	*p*
7	Longest run	0.2945	*p*	0.7785	*p*	0.5604	*p*
8	Non-overlapping template **	145/148	– ***	145/148	–	147/148	–
9	Overlapping template	0.6729	*p*	0.7003	*p*	0.8515	*p*
10	Random excursions **	8/8	–	8/8	–	8/8	–
11	Random excursion variant **	17/18	–	18/18	–	18/18	–
12	Rank	0.2574	*p*	0.3435	*p*	0.5741	*p*
13	Runs	0.6038	*p*	0.6626	*p*	0.9636	*p*
14	Serial test *p*-value 1	0.5463	*p*	0.086	*p*	0.2642	*p*
	Serial test *p*-value 2	0.4193	*p*	0.4075	*p*	0.041	*p*
15	Universal	0.293	*p*	0.7286	*p*	0.5081	*p*

* "*p*" denotes that the image passes the test at the significance level of 0.01. ** For the non-overlapping template, random excursions, and random excursion 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.

4.7. Computational Complexity

The computational complexity of the TCM-IEA mainly consists of three stages: key generation, bit-level permutation, and S-box-based feedback diffusion. In the first stage, the SHA-512 hash computation requires $O(MN)$ operations, where M and N denote the image dimensions. The subsequent parameter update and chaotic sequence generation involve linear iterations proportional to the image size, resulting in an overall complexity of $O(MN)$. In the bit-level permutation stage, the image is decomposed into eight bit-planes and processed through index-based rearrangement, circular shifts, and sorting operations. Since these operations involve linear traversal and local sorting within fixed-size structures, the overall complexity remains $O(MN)$. In the diffusion stage, each pixel undergoes S-box substitution and feedback-based XOR operations with chaotic sequences. As these operations are performed sequentially over all pixels, the computational complexity is also $O(MN)$. Therefore, the overall computational complexity of the TCM-IEA is $O(MN)$, which is consistent with most existing chaos-based IEAs.

5. Conclusions

The TCM-IEA combines a three-dimensional CM with a plaintext-related key generation mechanism to achieve a strong security performance. By integrating the SHA-512 hash function with parameter reconstruction, the TCM-IEA enhances the key sensitivity and establishes a strong dependency between the plaintext and the generated keys. The introduction of a dynamically generated S-box further improves nonlinearity, while the bit-level permutation and feedback diffusion mechanisms effectively strengthen the confusion and diffusion properties. The experimental results demonstrate that the TCM-IEA achieved excellent statistical characteristics, including a high information entropy, a low pixel correlation, and a strong resistance against differential attacks. In addition, the TCM-IEA shows robustness under various attack scenarios such as noise and data loss, confirming its reliability in practical applications. The overall computational complexity is $O(MN)$, indicating that the TCM-IEA is efficient and suitable for image encryption tasks.

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

Funding: This research is supported by the Science and Technology Planning Project of Gansu (No: 23JRRA894, 24JRRA243), the Humanities and Social Sciences Research Planning Fund of the Ministry of Education of China (No: 24YJAZH242), and the Open Foundation of Key Laboratory of Computing Power Network and Information Security (No: SKLCN-2023-02).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no 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. Syst.* **2024**, *54*, 5237–5247. [CrossRef]
3. Fridrich, J. Symmetric ciphers based on two-dimensional chaotic maps. *Int. J. Bifurc. Chaos* **1998**, *8*, 1259–1284. [CrossRef]
4. Sarra, B.; Sun, H.; Dua, M.; Dua, S.; Dhingra, D. A novel 1D powered Chebyshev quadratic map-based image encryption using dynamic permutation-diffusion. *Sci. Rep.* **2026**, *16*, 9469. [CrossRef]
5. 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]
6. Vismaya, V.S.; Muni, S.S.; Panda, A.K.; Mondal, B. Degn-Harrison map: Dynamical and network behaviours with applications in image encryption. *Chaos Solitons Fractals* **2025**, *192*, 115987.
7. Nazish, M.; Banday, M.T. A novel Fibonacci-Sequence-based chaotification model for enhancing chaos in 1-D maps. *IEEE Internet Things J.* **2025**, *11*, 40268–40277. [CrossRef]
8. Lin, Y.; Liao, Y.; Zeng, W.; Wei, Y.; Chen, D.; Yuan, X.; Li, Y.; Erkan, U.; Toktas, A.; Zhang, C.; et al. 3D non-degenerate hyperchaos: Design, analysis, and application in image encryption. *IEEE Trans. Consum. Electron.* **2026**. [CrossRef]
9. Liao, Y.; Lin, Y.; Xing, Z.; Li, Q.; Huang, G.; Chen, D.; Yuan, X. Using 3D-LMM-based encryption to secure digital images with 3D S-box and Fibonacci Q-Matrix. *IEEE Internet Things J.* **2025**, *12*, 55182–55195. [CrossRef]
10. 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]
11. Gao, S.; Ding, S.; Iu, H.H.-C.; Erkan, U.; Toktas, A.; Simsek, C.; Wu, R.; Xu, X.; Cao, Y.; Mou, J. A three-dimensional memristor-based hyperchaotic map for pseudorandom number generation and multi-image encryption. *Chaos Interdiscip. J. Nonlinear Sci.* **2025**, *35*, 073105. [CrossRef]
12. Lin, Y.; Wei, Y.; Chen, D.; Li, Y.; Erkan, U.; Toktas, A.; Gao, S.; Zhang, Y. Cryptanalysis and Improvement of a Video Cryptosystem via Chaos and S-Box. *ACM Trans. Multimed. Comput. Commun. Appl.* **2026**. [CrossRef]

13. 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]
14. Jiang, D.; Ni, J.; Alasbali, N.; Nafea, I.; Liu, Q.; Ahmad, J.; Boulila, W. DDCM: Cracking anonymized facial images using denoising diffusion cryptanalytic model. *IEEE Trans. Consum. Electron.* **2025**, *71*, 4464–4474. [CrossRef]
15. Lai, Q.; Miao, B. A fast multi-image encryption scheme using hyperchaotic map and parallel algorithm for vehicle detection applications. *IEEE Internet Things J.* **2026**, *13*, 19590–19599. [CrossRef]
16. Kareem, S.; Pirdawood, M. Algebraically enhanced 3D chaotic map with hash-based initialization for secure image encryption. *Theory Comput. Syst.* **2026**, *70*, 30. [CrossRef]
17. Aibai, A.; Nuermaimaiti, M.; Tuersun, Y.; Ghopur, D. A novel 2D hyperchaotic map for secure financial data encryption. *Entropy* **2026**, *28*, 262. [CrossRef] [PubMed]
18. Duan, Y.; Xu, X.; Banerjee, S.; Cao, Y.; Mou, J. A multi-image encryption scheme for gray-color images based on 2D chaotic map and dual S-boxes. *Nonlinear Dyn.* **2026**, *114*, 249. [CrossRef]
19. Deepankumar, S.; Rengaraj, R.; Pranesh, R.; Mishra, P.; Pal, A.K. A 3D chaotic map-based novel intra and inter-level bit plane image content encryption. *J. Inf. Secur. Appl.* **2026**, *98*, 104389.
20. Kocak, O.; Erkan, U.; Babaoglu, I. Bit-level quad-block shuffling and sequential summing dispersing image encryption based on hyperchaotic 2D Euler Pi crossed Sine map. *Digit. Signal Process.* **2026**, *173*, 105941. [CrossRef]
21. Liu, R.; Liu, H.; Zhao, M. Cryptanalysis and construction of keyed strong S-Box based on random affine transformation matrix and 2D hyper chaotic map. *Expert Syst. Appl.* **2024**, *252*, 124238. [CrossRef]
22. Sun, S.; Yang, W.; Yin, Y.; Tian, X.; Li, G.; Deng, X. A color image encryption scheme utilizing a logistic-sine chaotic map and cellular automata. *Sci. Rep.* **2025**, *15*, 21603. [CrossRef]
23. Gao, J.; Teng, L. A chaotic multi-image encryption scheme based on block space jump scrambling and dynamic sliding queue diffusion. *Nonlinear Dyn.* **2025**, *113*, 31691–31723. [CrossRef]
24. Wang, H.; Lai, Q. Exploiting Compressed Sensing and DNA Coding for multi-image encryption scheme with application to industrial internet of things. *IEEE Trans. Ind. Electron.* **2025**, *72*, 14543–14552. [CrossRef]
25. Wang, X.; Liu, X.; Peng, J. A novel image encryption scheme based on a 3D enhanced chaotic map and DNA computing model. *J. Frankl. Inst.* **2025**, *362*, 107790. [CrossRef]
26. Liu, S.; Ye, X. Designing a novel image encryption scheme based on an improved 2D logistic map. *J. Appl. Phys.* **2024**, *136*, 124304. [CrossRef]
27. Su, Y.; Teng, L.; Liu, P.; Unar, S.; Wang, X.; Fu, X. Visualized multiple image selection encryption based on Log chaos system and multilayer cellular automata saliency detection. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 4689–4702. [CrossRef]
28. Toktas, A.; Erkan, U.; Gao, S.; Pak, C. A robust bit-level image encryption based on Bessel map. *Appl. Math. Comput.* **2024**, *462*, 128340. [CrossRef]
29. 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]
30. Maity, A.; Dhara, B.C. An audio encryption scheme based on empirical mode decomposition and 2D Cosine Logistic map. *IEEE Lat. Am. Trans.* **2024**, *22*, 267–275. [CrossRef]
31. Jackson, J.; Perumal, R. A robust medical image encryption technique using inverse cosine chaotic map. *Expert Syst. Appl.* **2026**, *298*, 129574.
32. 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]
33. Özpolat, E.; Çelik, V.; Gülten, A. Hyperchaotic system-based PRNG and S-box design for a novel secure image encryption. *Entropy* **2025**, *27*, 299. [CrossRef]
34. Qiu, D.; Zhang, T.; Liu, J.; Liu, S.; He, P. An innovative image encryption algorithm based on the DNAS_box and hyperchaos. *Entropy* **2025**, *27*, 239. [CrossRef] [PubMed]
35. Sun, M.; Yuan, J.; Li, X.; Liu, D.; Wei, X. LVGG-IE: A novel lightweight VGG-based image encryption scheme. *Entropy* **2024**, *26*, 1013. [CrossRef]
36. Si, Y.; Liu, H.; Chen, Y. Constructing Keyed Strong S-Box Using an Enhanced Quadratic Map. *Int. J. Bifurc. Chaos* **2021**, *31*, 2150146. [CrossRef]
37. Liu, H.; Kadir, A.; Xu, C. Cryptanalysis and constructing S-Box based on chaotic map and backtracking. *Appl. Math. Comput.* **2020**, *376*, 125153. [CrossRef]
38. Faheem, Z.B.; Ali, A.; Khan, M.A.; Ul-Haq, M.E.; Ahmad, W. Highly dispersive substitution box (S-Box) design using chaos. *Electron. Telecommun. Res. Inst. J.* **2020**, *42*, 619–632.
39. El-Latif, A.A.A.; Abd-El-Atty, B.; Amin, M.; Iliyasu, A.M. Quantum-inspired cascaded discrete-time quantum walks with induced chaotic dynamics and cryptographic applications. *Sci. Rep.* **2020**, *10*, 1930. [CrossRef]

40. Liu, X.; Zheng, S.; Yang, J. Color image encryption scheme based on a novel 2D-CLCM chaotic system and RNA encoding. *Math. Comput. Simul.* **2026**, *239*, 340–360. [CrossRef]
41. Devi, C.S.; Amirtharajan, R. A novel 2D MTMHM based key generation for enhanced security in medical image communication. *Sci. Rep.* **2025**, *15*, 25411. [CrossRef]
42. Obaid, M.J.; Neamah, A.A.; Shukur, A.A.; Pham, V.T.; Grassi, G. A reliable color image encryption scheme based on a novel dual-wing hyperchaotic map. *Expert Syst. Appl.* **2025**, *289*, 128237. [CrossRef]
43. Lin, Z.; Liu, H. Constructing a non-degeneracy 3D hyperchaotic map and application in image encryption. *Multimed. Tools Appl.* **2024**, *83*, 82049–82068. [CrossRef]
44. 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]
45. Liu, P.; Wang, X.; Su, Y.; Liu, H.; Unar, S. Globally coupled private image encryption algorithm based on infinite interval spatiotemporal chaotic system. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2023**, *70*, 2511–2522. [CrossRef]
46. Liu, X.; Sun, K.; Wang, H. A novel image encryption scheme based on 2D SILM and improved permutation-confusion-diffusion operations. *Multimed. Tools Appl.* **2023**, *82*, 23179–23205. [CrossRef]
47. Su, Y.; Teng, L.; Yan, X.; Sun, C.; Xian, Y. Spatiotemporal chaos in a recursively permuted lattice with dual maps on odd and even sites and its application in image encryption. *Phys. Scr.* **2026**, *101*, 075203. [CrossRef]

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.