



Article

Fractal–Chaos-Based Controllable Compressive Sensing and Adaptive Matching Embedding for Multi-Image Encryption and Concealment

Chao Wang , Zhao Li *, Weijun Cheng and Yucong Lai

College of Mathematics and Computer, Guangdong Ocean University, Zhanjiang 524088, China; wangchao@gdou.edu.cn (C.W.)

* Correspondence: zhaoli@gdou.edu.cn

Abstract

In this paper, a multi-image encryption and hiding scheme based on controllable compressive sensing and adaptive matching embedding is proposed. The highlights of this scheme include the use of adaptive data segmentation, controllable integer compressive sensing, and adaptive matching embedding. Firstly, a 512-bit master key, together with an HMAC-SHA512-based session key derivation mechanism incorporating plaintext hashes and random noise, is used to initialize the proposed Fractal–Chaos Hybrid Map—a novel chaotic system combining the Lorenz system with Mandelbrot fractal perturbations. This produces high-quality random sequences with superior randomness and a large key space. Secondly, the scheme applies DCT and adaptive data segmentation to process the plain images. The sparse data obtained from segmentation is compressed using a sensing matrix generated from random sequences, and then encoded and encrypted via random flipping. Finally, the adaptive matching embedding technique is used to embed the encrypted data into the carrier image. The proposed scheme performs excellently in both the compression encryption of plain images and the embedding of encrypted data into the carrier image. It supports the encryption and hiding of up to six plain images with an acceptable trade-off between embedding capacity and carrier image quality. A comprehensive security analysis, including statistical and differential attack evaluations, confirms the scheme's strong resistance to various cryptographic threats while maintaining high efficiency.

Keywords: image hiding; adaptive data segmentation; chaotic map; controllable integer compressive sensing



Received: 16 June 2026

Revised: 24 July 2026

Accepted: 26 July 2026

Published: 28 July 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

In the era of information explosion, digital images have become a widely used multimedia data type. However, with the rapid development of information and communication technology, digital images are increasingly vulnerable to severe security threats. To protect the confidentiality and integrity of image data, image encryption and information hiding have become hot research areas. The goal of image encryption is to transform image data into an unintelligible form to prevent unauthorized access [1–3]. On the other hand, information hiding focuses on concealing secret information within an image, making it visible only to specific recipients while having little or no impact on the quality of the carrier image [4–6]. In the field of information hiding, many steganographic algorithms and techniques are available for embedding secret information into images. These methods typically

exploit hidden space within the image to embed secret information into pixel values or other image features. Common information hiding techniques include Least Significant Bit (LSB) steganography, Discrete Cosine Transform (DCT) steganography, and others [7–10]. Additionally, there are methods that combine image encryption and information hiding, allowing for the protection of image data confidentiality while also hiding secret information within the image. These methods are particularly useful in specific application scenarios, such as digital watermarking for image authentication and copyright protection. Optical watermarking techniques [5] offer advantages in parallel high-speed processing. Reversible data hiding in encrypted images (RDHEI) [11,12] enables error-free data extraction and lossless image recovery, with recent advances achieving high embedding capacity and CPA-security. Coverless information hiding [13] avoids carrier modification entirely by retrieving visually similar images, enhancing undetectability. Hybrid frameworks combining chaotic encryption with compressive sensing [14] achieve efficient multi-image hiding, while deep learning-based watermarking [15] improves robustness against various attacks.

Image encryption has seen significant development, and various excellent schemes have been proposed. However, the majority of these schemes directly transmit the plain image as the ciphertext image/data, which can easily be identified and attacked by adversaries, thereby increasing the risk. Encrypting the plain image and embedding it into a carrier image, making the carrier image visually indistinguishable from the unembedded encrypted image, can easily evade data filtering by attackers and greatly reduce the risk of being attacked. In recent years, researchers have gradually focused on studying such schemes. In 2015, Bao et al. introduced a new concept of image encryption, transforming the original image into visually meaningful encrypted images [16]. As an example of implementing this concept, the paper describes an image encryption system. Simulation results and security analysis demonstrate the outstanding encryption performance of the proposed concept and system. However, this scheme requires a large image as the carrier image. Subsequently, the concept of compressive sensing in image encryption was introduced. Compressive sensing, combined with transform domain quantization and coding, is a commonly used data compression technique in image compression encryption [16]. Chai et al. proposed an efficient, visually meaningful image encryption scheme that combines compressive sensing and LSB embedding [17]. The paper indicates that using LSB embedding allows for independent selection of the carrier image, without being influenced by the plain image [17]. Zhu et al. conducted related research by combining block compressive sensing and singular value decomposition embedding [18]. Their work aims to achieve a balance between security, compression performance, robustness, and operational efficiency [18]. The above-mentioned approaches all focus on encrypting and hiding information within a single image, with limited capacity to carry information. Combining chaotic systems, compressive sensing, and three-dimensional discrete cosine transform, Wang et al. proposed a novel three-image encryption and hiding scheme [19]. Subsequently, they proposed a chaotic system, a parallel compressive sensing scheme, and a fast Fourier-transform dual-image visually meaningful encryption scheme [14]. In 2023, Guan et al. proposed a novel reversible neural network-based framework for multi-image hiding, capable of concealing up to four plain images [20]. Their method demonstrates excellent imperceptibility, security, and recovery accuracy across multiple datasets [20]. Meanwhile, significant research has also been devoted to steganography techniques. Pairwise prediction-error expansion [21,22] jointly encodes adjacent prediction errors to exploit inter-pixel correlations for improved capacity. Controlled contrast enhancement and wavelet-based methods [23–25] enhance visual quality and embedding efficiency. Adaptive embedding schemes using oriented gradients, particle swarm optimization, and Harris hawks optimization [26–28] further improve imperceptibility and security. More

recently, reversible data hiding in encrypted images has been advanced through reservoir computing-based fusion [6] and LSB-based GUI tools [10] for practical applications. From the concepts of image encryption and hiding to the related schemes of image encryption and hiding, there have been many research achievements. However, existing schemes still lack research on the controllability of the number of encrypted and hidden images, the controllability of image quality after decryption, the controllability of carrier image quality after embedding information, and the flexibility of the schemes. Moreover, there is room for further improvement in the efficiency of these schemes.

Building upon existing research, we propose a multi-image encryption and hiding scheme based on controllable compressive sensing and adaptive matching embedding. Firstly, the required random sequences for subsequent encryption are generated using chaotic mappings. Then, the plain images undergo Discrete Cosine Transform (DCT), quantization, adaptive segmentation, controllable integer compressive sensing, and coding. Next, the carrier images undergo a modulo operation, and the resulting outputs are matched and corrected against the encrypted data to be embedded. Finally, the processed ciphertext data is adaptively embedded into the lower significant bits of the processed carrier image to obtain visually meaningful encrypted images. This scheme exhibits several notable highlights. First, adaptive segmentation enables more rational data classification, significantly improving the efficiency of compressive sensing (CS) processing and the overall compression ratio. The CS framework in Reference [29] employs cyclic measurement matrices to accelerate signal reconstruction, thereby substantially enhancing reconstruction efficiency. Building on this foundation, the proposed scheme introduces an efficient, controllable integer compressive sensing algorithm that further improves computational efficiency while enabling high-performance data compression.

More importantly, a Fractal–Chaos Hybrid Map is incorporated as a core component of the encryption mechanism, which constitutes a key innovation of this work. By integrating a modified Lorenz chaotic map with Mandelbrot-based fractal perturbations, the proposed hybrid map generates high-entropy, strongly unpredictable pseudorandom sequences with enhanced sensitivity to initial conditions and control parameters. This significantly strengthens the security of the encryption process and expands the effective key space. Furthermore, adaptive modulo operations and adaptive matching embedding strategies allow a larger amount of encrypted information to be embedded into the carrier image, improving the visual quality of visually meaningful encrypted images while reducing data-level discontinuities. As a result, the scheme effectively lowers the risk of statistical detection and cryptanalytic attacks. Finally, the overall framework supports partial decryption to obtain a coarse preview of the encrypted image, thereby improving flexibility and expanding the practical applicability of the proposed scheme.

2. Multi-Image Encryption and Hiding Scheme

In this section, we will introduce and analyze each component of the scheme, provide a detailed overview of the entire scheme, and present the scheme's flowchart. The main techniques employed in this scheme include improved chaotic mappings and key distribution for ensuring the security of the scheme. The processing of plain images involves DCT, quantization, adaptive data segmentation, and controllable integer compressive sensing and random binary flipping. The processing of the carrier image involves data encryption and embedding based on modulo operations and adaptive matching correction.

2.1. Fractal–Chaos Hybrid Map and Key Distribution

Chaotic mappings are widely used in image encryption due to their sensitive dependence on initial conditions and parameter variations. However, chaotic sequences

generated from purely polynomial maps may still exhibit statistical non-uniformity or detectable structural patterns, which can undermine cryptographic security. To address these limitations, we propose the Fractal–Chaos Hybrid Map (FCHM), which combines a modified Lorenz map with a non-algebraic Mandelbrot perturbation to enhance randomness and resist statistical attacks.

2.1.1. Construction and Rationale of the Fractal–Chaos Hybrid Map

In this part, we formally define the proposed Fractal–Chaos Hybrid Map, which combines a modified Lorenz chaotic map with a fractal perturbation derived from the Mandelbrot iteration.

The Mandelbrot escape-time function $f_M(x, y)$ is non-algebraic: it admits no closed-form expression and is computable only through iterative evaluation. When this function is injected into the Lorenz dynamics, the resulting state update becomes a non-algebraic mapping that resists the two principal classes of attacks on chaotic ciphers:

1. **Parameter estimation attacks.** For a pure polynomial chaotic map (such as the Lorenz map), an attacker who observes the output sequence can, in principle, set up a system of equations to estimate the secret parameters. The Mandelbrot perturbation introduces a term that cannot be expressed in closed form, making this inverse problem computationally intractable.
2. **Phase-space reconstruction.** The Lorenz attractor possesses a well-characterized geometric structure that can be exploited for attack. The Mandelbrot perturbation acts as a state-dependent, non-algebraic injection that disrupts this deterministic structure, yielding a phase-space distribution that is significantly harder to model.

The NIST results reported in the ablation study (see Section 3.2.2) provide empirical validation of this theoretical argument: the perturbation does not merely add complexity for its own sake, but systematically eliminates statistical defects that the standalone Lorenz map exhibits.

Standard PRNGs are designed to pass statistical tests but remain deterministic finite-state machines whose algorithms are publicly known. If the internal state can be recovered, all past and future outputs can be predicted. The non-algebraic Mandelbrot component makes the internal state significantly harder to recover from observed output. We note that the hybrid map is not intended as a general-purpose replacement for standard PRNGs, but rather as a specialized chaotic source for image encryption where the expanded key space and resistance to dynamical reconstruction provide concrete security advantages.

Definition 1 (Mandelbrot Iteration). *Given a complex variable*

$$c = x + iy \in \mathbb{C},$$

define the iterative sequence

$$z_0 = 0, \quad z_{n+1} = z_n^2 + c, \quad n = 0, 1, \dots, N-1. \quad (1)$$

The divergence index is defined as

$$n_f = \min\{n : |z_n| > 2\}, \quad (2)$$

and if the above set is empty, let $n_f = N$. We denote the final value by $z_f = z_{n_f}$.

Definition 2 (Modified Lorenz Chaotic Map). Given parameters $a, b, c, k > 0$ and a unit time step $\Delta t = 1$, the modified discrete Lorenz map updates the state (x_n, y_n, z_n) to $(x_{n+1}, y_{n+1}, z_{n+1})$ as follows:

$$\begin{aligned}x_{n+1} &= x_n + a(y_n - x_n) + k \sin(2\pi x_n), \\y_{n+1} &= y_n + x_n(b - z_n) - y_n, \\z_{n+1} &= z_n + x_n y_n - c z_n.\end{aligned}\quad (3)$$

For notational convenience in the hybrid construction, we define the chaotic increments as

$$\begin{aligned}D_x(x_n, y_n, z_n) &= a(y_n - x_n) + k \sin(2\pi x_n), \\D_y(x_n, y_n, z_n) &= x_n(b - z_n) - y_n, \\D_z(x_n, y_n, z_n) &= x_n y_n - c z_n,\end{aligned}\quad (4)$$

so that the state update can be written compactly as $(x_{n+1}, y_{n+1}, z_{n+1}) = (x_n, y_n, z_n) + (D_x, D_y, D_z)$. The sinusoidal coupling term $k \sin(2\pi x_n)$ introduces nonlinear cross-coupling that enhances the chaoticity of the discrete trajectory compared to the classical forward-Euler discretization of the Lorenz system.

Definition 3 (Fractal Perturbation Weight). Let $\gamma > 0$ denote the fractal perturbation strength. The weight generated from the Mandelbrot iteration is given by

$$w = \gamma \frac{n_f}{N}. \quad (5)$$

The complex fractal output can be decomposed as

$$z_f = \Re(z_f) + i\Im(z_f). \quad (6)$$

Definition 4 (Fractal–Chaos Hybrid Map). Given the current state $(x_n, y_n, z_n) \in [0, 1]^3$, the hybrid map $F : [0, 1]^3 \rightarrow [0, 1]^3$ produces the next state $(x_{n+1}, y_{n+1}, z_{n+1})$ as follows:

$$\begin{aligned}x_{n+1} &= (x_n + D_x(x_n, y_n, z_n) + w \Re(z_f)) \bmod 1, \\y_{n+1} &= (y_n + D_y(x_n, y_n, z_n) + w \Im(z_f)) \bmod 1, \\z_{n+1} &= (z_n + D_z(x_n, y_n, z_n) + \frac{w}{2} |z_f|) \bmod 1,\end{aligned}\quad (7)$$

where D_x, D_y, D_z are the Lorenz increments from (4), w is the fractal weight from (5), and $\Re(z_f), \Im(z_f)$ are the real and imaginary parts of the final Mandelbrot iterate from (2).

Proposition 1 (Boundedness of the Hybrid Map). For any initial condition $(x_0, y_0, z_0) \in [0, 1]^3$, the hybrid map satisfies

$$F(x_0, y_0, z_0) \in [0, 1]^3.$$

Proof. The modulo operation in (7) ensures that each component of $F(x, y, z)$ lies within the interval $[0, 1)$. $\square$

2.1.2. Dynamical Analysis and Chaos Verification

To comprehensively evaluate the chaotic properties of the Fractal–Chaos Hybrid Map and its suitability for cryptographic applications, a series of dynamical indicators are examined, including the Lyapunov exponent, bifurcation diagram, phase portrait, Permutation Entropy (PE), Sample Entropy (SE), Phase-Space Coverage (Coverage), and Kolmogorov–Smirnov uniformity distance (KS).

The Lyapunov exponent spectra for all five control parameters are shown in Figure 1. Across the entire parameter ranges, all three exponents remain strictly positive, confirming that the FCHM consistently operates in a hyperchaotic regime. For a and b , LE_3 stays near 1 while LE_1 and LE_2 rise monotonically from roughly 1.7 and 1.5 to about 2.7 and 3.6, respectively, showing that larger values strengthen the chaotic instability. Parameter c induces a chaos-to-hyperchaos transition: LE_3 crosses zero at $c \approx 1$ and reaches approximately 1.7, whereas LE_1 and LE_2 remain nearly constant at around 3.1 and 2.6. The exponents are largely insensitive to k and γ , stabilizing near 3.2, 2.5, and 1.0, which demonstrates the robustness of the hyperchaotic dynamics.

The bifurcation diagram (Figure 2) exhibits a fully saturated, space-filling point distribution across the entire $[0, 1]$ phase space, visually confirming the absence of periodic windows and the establishment of global chaos. This ergodic behavior originates from the synergistic interaction among Lorenzian mixing, Mandelbrot-set perturbation, and the modular folding operation.

The three-dimensional phase trajectory (Figure 3a) shows that successive iterates continuously populate arbitrary locations throughout the unit cube, indicating strong ergodicity. The corresponding phase density maps (Figure 3b–d) reveal a complex intermingling of high- and low-probability regions with no recognizable pattern. In the X–Y projection, no functional dependence is discernible, implying that the state variables evolve with a high degree of mutual independence. A slightly elevated density of low-probability spots appears in the X–Z and Y–Z planes within $z \in [0.2, 0.4]$, reflecting a mild anisotropy in the invariant measure of the z -component. This non-uniformity is attributed to the asymmetric dynamical role of z as a fractal-modulated perturbation term. Importantly, the overall distribution remains sufficiently irregular and space-filling, ensuring that no exploitable statistical bias is introduced into the generated keystream.

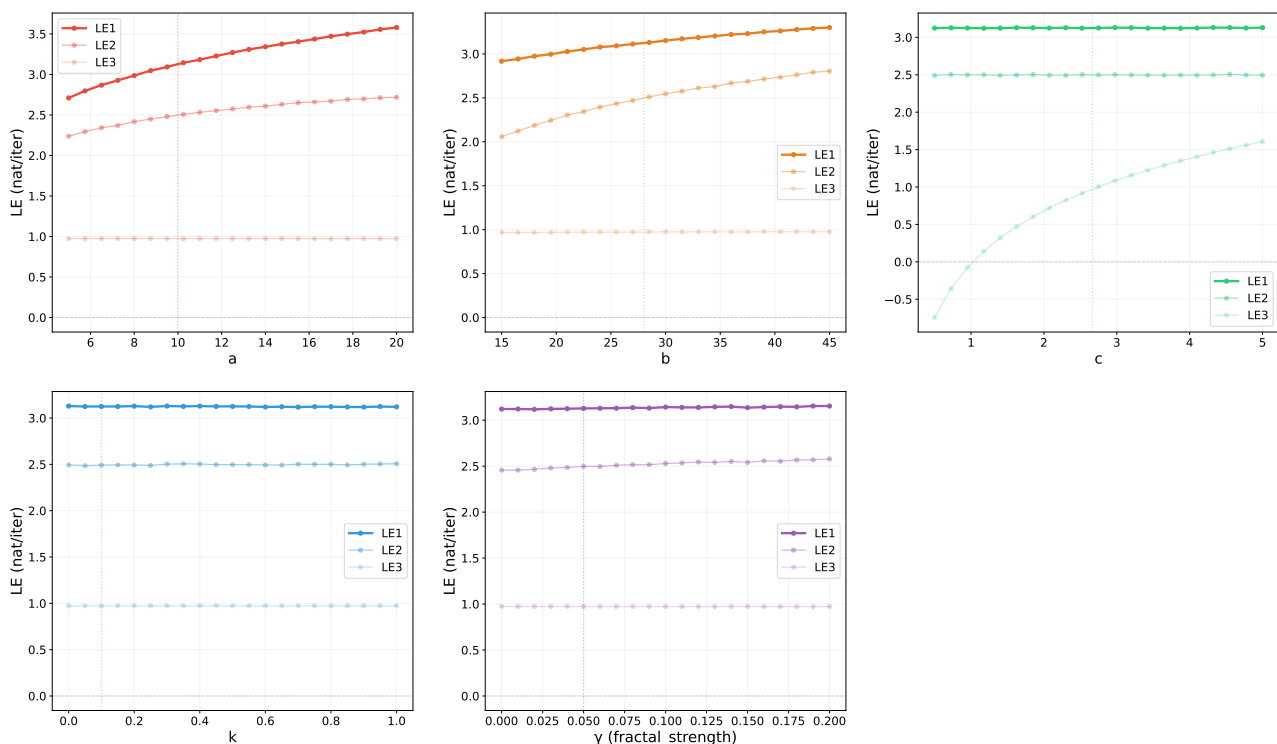


Figure 1. Lyapunov exponent spectra of the FCHM for all five control parameters.

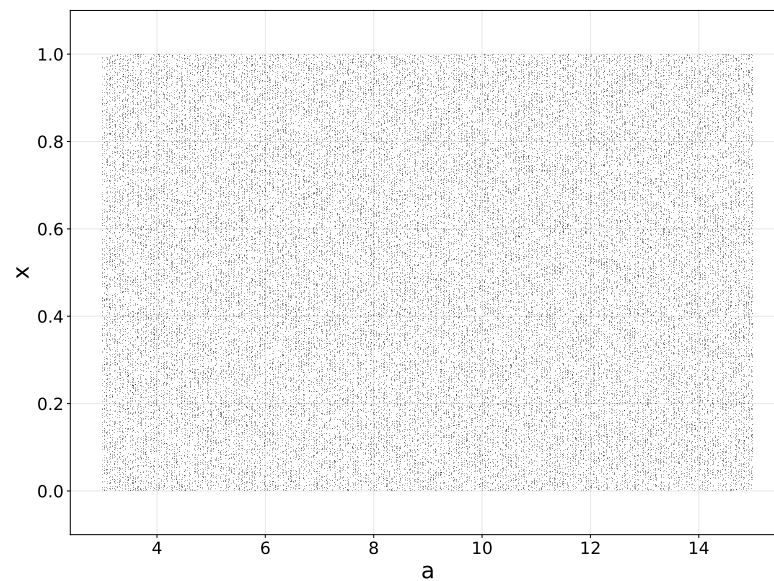


Figure 2. Bifurcation diagram of the FCHM with respect to parameter a (x -channel).

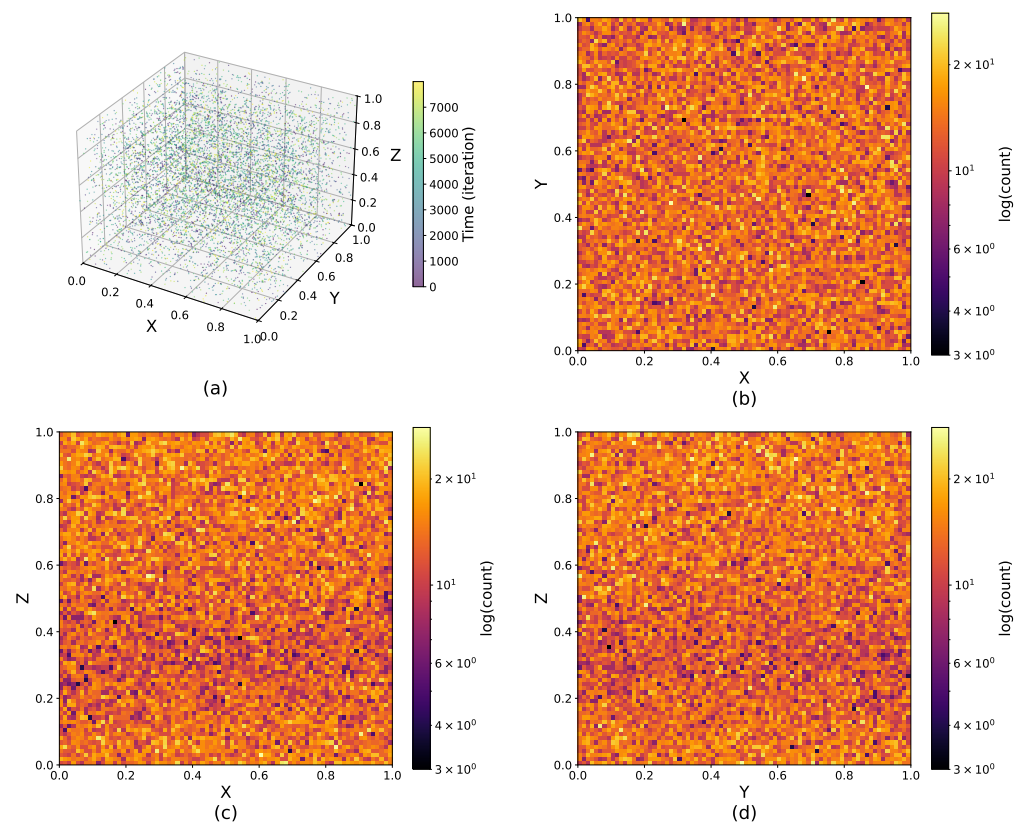


Figure 3. Phase-space analysis of the FCHM: (a) 3D phase, (b–d) phase density maps in the X – Y , X – Z , and Y – Z planes.

Table 1 compares PE, SE, Coverage, and KS distance of several chaotic maps (2P-ELCM [30], LPWLCM [31], FracQ Logistic–Tent [32], Quantum Logistic [33], CCS [34], Chua Multi–Scroll [35], 4D Hyperchaos [36]). FCHM achieves a PE of 0.9983, the highest among all tested systems, and an SE of 2.1887, second only to 2.2025 of 2P-ELCM but well above the remaining maps. The phase-space coverage reaches 1.0000, indicating that the entire state space is fully explored, and the KS distance is 0.0196, confirming near-uniform output distribution. FCHM and 2P-ELCM rank in the top tier of chaotic complexity,

significantly outperforming the remaining maps, including CCS, FracQ Logistic–Tent, Chua Multi–Scroll, and 4D Hyperchaos. These results demonstrate that FCHM generates state sequences with high structural richness and near-ideal phase-space utilization, providing a solid foundation for cryptographic keystream generation.

Table 1. Comparison of complexity and statistical metrics for different chaotic maps.

#	System	PE	SE	Coverage	KS
1	FCHM (Proposed)	0.9983	2.1887	1.0000	0.0196
2	2P-ELCM	0.9977	2.2025	1.0000	0.0101
3	LPWLCM	0.9967	2.1304	1.0000	0.0485
4	Quantum Logistic	0.9938	2.0140	1.0000	0.1087
5	CCS	0.8815	1.0731	1.0000	0.1473
6	FracQ Logistic–Tent	0.7560	0.6104	0.3400	0.7093
7	Chua Multi–Scroll	0.2342	0.0739	1.0000	0.2198
8	4D Hyperchaos	0.1636	0.0506	0.3467	0.7952

2.1.3. Key Hierarchy and Parameter Mapping

In the proposed scheme, a two-tier key hierarchy is employed to eliminate the need for per-message secure channels while maintaining plaintext-dependent encryption.

A 512-bit master key K_{master} is shared between the communicating parties during system initialization. This can be accomplished via a standard key exchange protocol (e.g., Diffie–Hellman) or through a pre-installed secure hardware module. The master key is long-term and does not change per encryption session.

For each plain image P , the sender computes its SHA-512 hash value:

$$H = \text{SHA-512}(P). \quad (8)$$

The 512-bit session key K_{session} is then derived using HMAC-based key derivation:

$$K_{\text{session}} = \text{HMAC-SHA512}(K_{\text{master}}, H \parallel 0x01), \quad (9)$$

where $\parallel$ denotes concatenation and $0x01$ is a domain separator. The hash value H is transmitted alongside the ciphertext as auxiliary data (it does not need to be encrypted, as knowledge of H alone does not reveal K_{session} without K_{master}).

Upon receiving the ciphertext, the receiver extracts H , retrieves the pre-shared K_{master} , and applies the identical HMAC-SHA512 derivation to obtain the same K_{session} . No additional secure channel is required beyond the initial master key exchange.

Let K_{session} denote the derived 512-bit key, represented as a large integer. The key is partitioned into segments corresponding to the parameter set

$$\Theta = \{x_0, y_0, z_0, a, b, c, k, \gamma, N\}.$$

Each segment consists of B_p bits, with the total satisfying $\sum_{p \in \Theta} B_p = 512$. Denote the integer value of each segment by s_p , and let $D_p = 2^{B_p} - 1$ be the maximum representable value in the segment.

Continuous Parameters

For parameters $p \in \{x_0, y_0, z_0, a, b, c, k, \gamma\}$ with range $[p_{\min}, p_{\max}]$, the mapping from the key segment s_p to the parameter value is defined as

$$p = p_{\min} + \frac{s_p}{D_p}(p_{\max} - p_{\min}). \quad (10)$$

Integer Parameter

For the integer parameter $N \in [N_{\min}, N_{\max}]$, the mapping is given by

$$N = \text{clamp} \left(\left\lfloor N_{\min} + \frac{s_N}{D_N} (N_{\max} - N_{\min}) + 0.5 \right\rfloor, N_{\min}, N_{\max} \right), \quad (11)$$

where $\lfloor \cdot \rfloor$ denotes the floor operation, the $+0.5$ term implements rounding to the nearest integer, and $\text{clamp}(\cdot)$ ensures that N remains within the specified bounds.

Bit Extraction Order

The segments s_p are extracted from K_{session} in a fixed order (from the most significant bit to the least significant bit) according to the allocated bit lengths B_p . This procedure ensures a reproducible mapping from the 512-bit session key to the complete set of system parameters.

Parameter Ranges

The recommended ranges are as follows:

$$\begin{aligned} x_0, y_0, z_0 &\in [0, 1), \quad a \in [8, 18], \quad b \in [20, 40], \quad c \in [2, 4], \\ k &\in [0, 1], \quad \gamma \in [0, 0.2], \quad N \in \{20, 21, \dots, 83\}. \end{aligned}$$

These ranges are selected based on Lyapunov exponent and bifurcation analyses (Section 2.1) and correspond to regions of robust chaotic behavior with consistently positive Lyapunov exponents and no periodic windows. The ranges are chosen conservatively with safety margins to ensure that numerical fluctuations do not compromise chaoticity. Equations (10) and (11) provide a deterministic and reproducible method for generating all system parameters from the 512-bit session key K_{session} .

2.2. DCT and Adaptive Data Segmentation

Regarding JPEG (Joint Photographic Experts Group), it is a widely used image compression standard jointly developed by ISO and IEC. In the JPEG compression process, the image is first divided into 8×8 non-overlapping blocks. Each block undergoes a DCT (Discrete Cosine Transform), followed by quantization. The quantization step divides each DCT coefficient by the corresponding element in a fixed quantization table and rounds the result to the nearest integer:

$$D(i, j) = \text{round} \left(\frac{\text{DCT}(A)(i, j)}{Q_Y(i, j)} \right). \quad (12)$$

In the standard JPEG luminance quantization table Q_Y , smaller step sizes are assigned to low-frequency components (upper-left region) to preserve the overall visual structure of the image, while larger step sizes are assigned to high-frequency components (lower-right region). This forces many high-frequency coefficients to zero after rounding, thereby achieving effective compression.

We randomly select an 8×8 image block A (Equation (13)). After applying DCT and quantizing with Q_Y , the resulting coefficient matrix D is shown in Equation (14).

$$A = \begin{bmatrix} 170 & 170 & 163 & 161 & 158 & 156 & 149 & 148 \\ 169 & 169 & 157 & 160 & 155 & 150 & 149 & 147 \\ 167 & 160 & 161 & 161 & 154 & 151 & 146 & 146 \\ 170 & 160 & 158 & 156 & 149 & 145 & 146 & 138 \\ 164 & 158 & 155 & 153 & 150 & 145 & 140 & 135 \\ 160 & 160 & 158 & 154 & 150 & 143 & 133 & 127 \\ 160 & 160 & 152 & 154 & 145 & 137 & 132 & 117 \\ 161 & 155 & 149 & 149 & 143 & 136 & 133 & 118 \end{bmatrix}, \quad (13)$$

$$D = \begin{bmatrix} 12 & 7 & -1 & 1 & 0 & 0 & 0 & 0 \\ 4 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (14)$$

As shown in D , the majority of high-frequency coefficients become zero after quantization, which is consistent with the intended compression effect of JPEG.

To enhance data security and compression effectiveness, the scheme introduces Compressive Sensing (CS). Before encoding, the data is processed with CS. In JPEG algorithm, when the image compression quality index is large, the proportion of non-zero values in the quantized data is high, which is not conducive to CS processing. In JPEG, the quantized data is transformed into a 1D data using the zigzag method. The first element is extracted as the DC component, while the remaining data represents the AC components. DC components usually have large numerical values. Therefore, when encoding them, a differential processing is applied, as shown in Equation (15).

$$DC_i = DC_i - DC_{i-1}, \text{ where } i = 2, 3, 4, \dots, \text{length}(DC). \quad (15)$$

This processing method indirectly improves coding efficiency and data compression effectiveness. Based on this idea and the characteristics of the data, we divide the transformed data into two parts: the Non-Sparse (NS) component and the Sparse (S) component. Simple partitioning can lead to an unstable proportion of non-zero elements in the sparse component, which in turn can cause reconstruction errors and low efficiency in CS processing. In this scheme, we use a threshold to divide the data, ensuring that the proportion of non-zero elements in the sparse component is not higher than the corresponding threshold values, which are $\frac{1}{32}$, $\frac{1}{16}$, $\frac{1}{8}$, and $\frac{1}{4}$ sequentially. This partitioning method guarantees a stable proportion of non-zero elements in the sparse component, ensuring more accurate and efficient reconstruction in compressive sensing. Additionally, this method allows for adaptive data segmentation based on security, compression, and functional requirements. For example, in the aforementioned example, if the adaptive segmentation parameter is chosen as $\frac{1}{16}$, the non-sparse component would be $[12, 7, 4]$, and the sparse component would be $[0, -1, -1, 1, 1, 0, \dots, 0]$. Similar to how JPEG can recover the coarse information of the original image using only the DC component, our method can recover coarse information using only the non-sparse component, with controllable information content. The more elements in the non-sparse component, the more information can be obtained, and the clearer the image will be.

2.3. Controllable Integer Compressive Sensing

Compressive Sensing theory states that if a signal is naturally or sparsely sparse in certain transform domains, then a high-dimensional signal can be projected onto a low-dimensional space using a measurement matrix that is independent of the sparse basis. These projections contain sufficient information to reconstruct the original signal by solving optimization problems associated with these projections with a high probability of success [37]. In previous image compression encryption schemes, the basic theory of compressive sensing is often used in a straightforward manner. However, this approach leads to low efficiency in signal compression and reconstruction. We propose the Controllable Integer Compressive Sensing (CICS) method, which utilizes an accelerated cyclic matrix. Both the sensing matrix and the signal are integers, resulting in improved computational efficiency. Additionally, during the reconstruction process, the threshold for solving the optimization problem is adjusted, significantly reducing the number of iterations and greatly improving the efficiency of signal reconstruction. The integerization of the sequence can be controlled as much as possible, meaning that the sensing matrix consists of only small integers or even just the values -1 , 0 , and 1 in the first M rows of a cyclic matrix. According to the properties of compressive sensing, this approach can control the range of compressed signal values, facilitating encoding and compression. The definition of the Controllable Integerization function, denoted as CInt , is as follows:

$$\text{CInt}(x) = \text{round}(\alpha_e x). \quad (16)$$

Here, α_e is the element proportion control parameter, where $\alpha_e \in (0, +\infty)$. x is a random sequence in the interval $(-1, 1)$.

The compressive sensing of a signal can be described as follows:

$$y = Ax, \quad (17)$$

where A is the sensing matrix, x is the sparse signal, and y is the measurement. Given y and A , the goal is to recover x . This signal recovery problem can be formulated as the following optimization problem:

$$\begin{aligned} & \arg\min_x \|x\|_0 \\ & \text{s.t. } \|y - Ax\|_2^2 \leq \varepsilon. \end{aligned} \quad (18)$$

Here, ε represents the reconstruction error. In this approach, the Orthogonal Matching Pursuit (OMP) method is used to solve the problem [38]. The description of this method is outlined in Algorithm 1. It is important to note that the sensing matrix A used for signal reconstruction consists of only a small number of integer elements in the first m rows of a cyclic matrix. Therefore, the matrix-vector multiplication operation that occurs during the signal reconstruction process can be accelerated using FFT. Both A and y are composed of integers, and in the reconstruction method, ε can be set to a suitable positive integer, such as $\varepsilon = 1$ in Step 7 of Algorithm 1. By relaxing the termination condition for the search of the solution, the algorithm can obtain the solution in very few iterations, greatly improving the computational efficiency of the algorithm. Unless otherwise stated, all experiments in this paper adopt $\varepsilon = 1$.

Different from the conventional OMP that terminates after a predetermined number of K iterations (where K is the assumed sparsity level), Algorithm 1 adopts a residual-driven stopping criterion. This design is motivated by the use of integer measurements in CICS, which allows a relaxed integer threshold (e.g., $\varepsilon = 1$) to be satisfied in very few iterations.

Algorithm 1 Orthogonal matching pursuit with residual-driven stopping.**Input:** measurement y , sensing matrix A , residual threshold ε

- 1: **Initialize** $k = 0, r^0 = y, S_0 = \emptyset$
- 2: $k = k + 1$
- 3: $s^k = \operatorname{argmax}_{1 \leq i \leq n} |\langle r^{k-1}, A_i \rangle|$
- 4: $S_k = S_{k-1} \cup \{s^k\}$
- 5: $\hat{x}_{S_k} = \operatorname{argmin} \|y - A_{S_k} x\|_2$
- 6: $r^k = y - A_{S_k} \hat{x}_{S_k}$
- 7: **Repeat** steps 2-6, until $|r^k| < \varepsilon$ {Residual-driven stopping}
- 8: $\hat{x} = \operatorname{argmin}_{x: \operatorname{supp}(x) = S_k} \|y - Ax\|_2$

Output: $\hat{x}$

Consequently, efficient reconstruction is achieved without requiring prior knowledge of the exact sparsity level K , which is often unavailable in practical scenarios.

2.4. Adaptive Embedding of Encrypted Data Based on Modulo Operation and Match Correction

The information in an image is mainly represented by the combination of different pixel values. In many cases, even if the numerical values are not exactly the same, reasonable pixel combinations can highlight the desired information of the image. Taking grayscale images as an example, we can convey image information in various forms. Due to the characteristics of human vision, subtle transformations of an image that are easy to overlook or difficult to perceive can also be used to secretly convey hidden information through seemingly identical images.

Each pixel in an image consists of 8 bits. The higher significant bits (HiSB) contain most of the energy information of the image, while the lower significant bits (LoSB) contain the detailed information. As shown in Figure 4, when the lower bits are gradually set to zero, the image details are gradually lost. However, when there are only a few low bits set to zero, it is difficult for the human eye to detect any abnormalities. This characteristic can be utilized by performing a modulo operation on the carrier image, removing the LoSB information, and embedding the cipher data into the lower bits of the image, as shown in Equation (19).

$$CaI_C = CaI - \operatorname{mod}(CaI, 2^n) + CiD. \quad (19)$$

Here, CaI represents the carrier image, CiD represents the cipher data, CaI_C represents the carrier image after embedding the cipher data, and n is the maximum number of bits required to store each data in the cipher data. The value of n is determined by the amount of data that needs to be embedded.

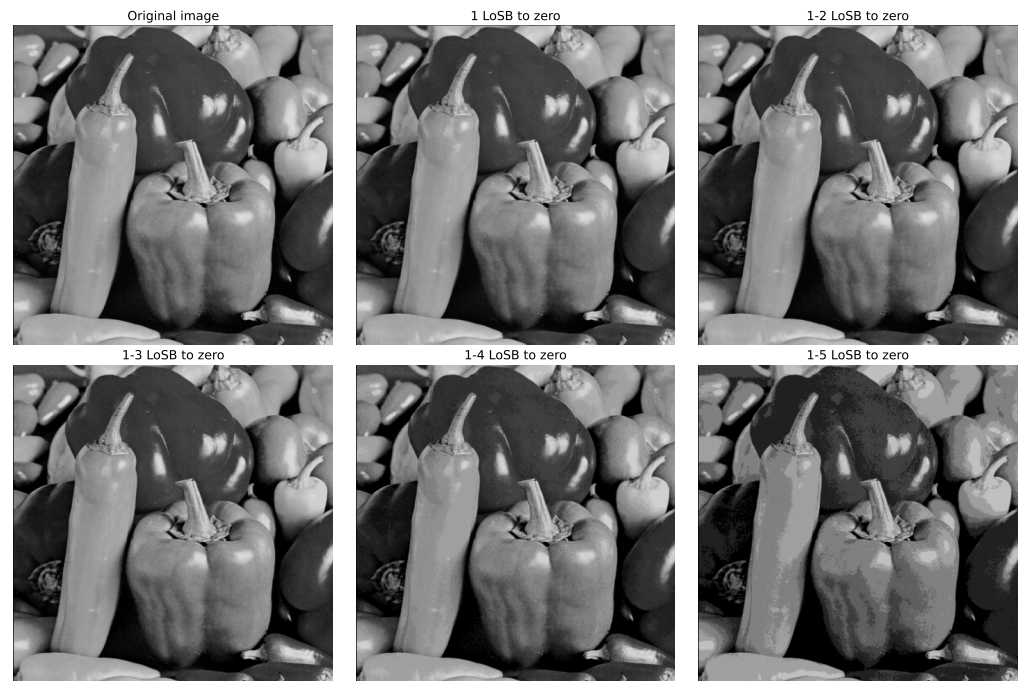


Figure 4. The standard test image ‘Peppers’ and its degradation after zeroing the lower significant bits.

In the embedding of cipher data, directly embedding irregular cipher data will result in slight noise characteristics in the image details due to the disruption of the LoSB. The lower bits will have obvious chaotic data features, making it susceptible to detection. Here, a match correction method is used to process the embedding data before embedding it.

Firstly, the data to be embedded is divided into three parts: the first part is used to store the encrypted data, the second part is for the original carrier image data, and the third part is for storing match correction information. Then, the LoSB of the carrier image ($\text{mod}(CaI, 2^n)$) and the embedding data (CiD) are divided into blocks of size $m \times m$. Random sequence index is generated based on the secret key to sort the embedding data for k times. Each time, the sorted data is compared with the LoSB of the carrier image, and the sorted data with the smallest sum of absolute differences is selected as the new embedding data. The corresponding sorting count $k_{i,j}$ is recorded.

Next, according to the requirements, the matching window is shifted to continue the match correction on the embedding data until satisfactory results are achieved. Finally, the matched and corrected final result is embedded into the carrier image. Figure 5 shows the diagram of adaptive embedding of encrypted data based on the modulo operation and match correction. The experimental results are shown in Figure 6, where the quality of the carrier image is improved after embedding and matching.

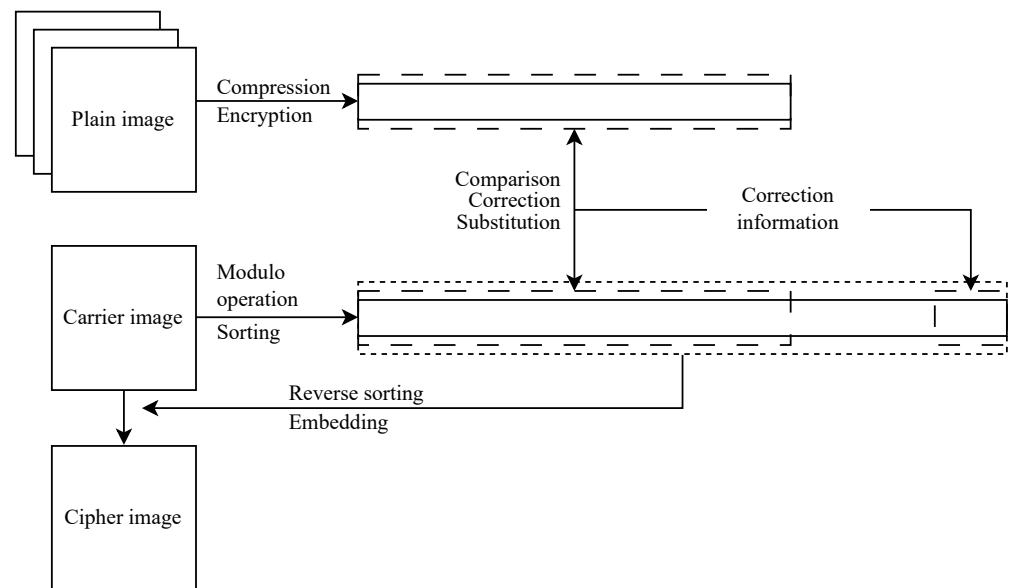


Figure 5. Diagram illustrating adaptive embedding of encrypted data based on modulo operation and match correction.

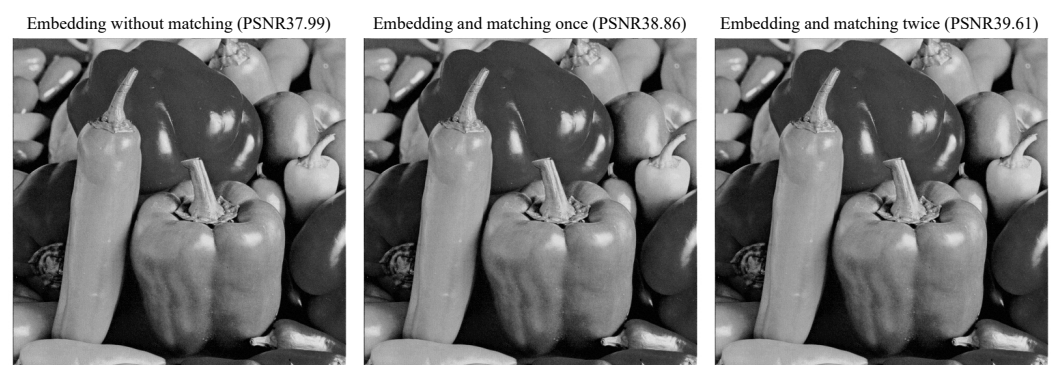


Figure 6. Experimental results of embedding and matching.

2.5. Whole-Image Encryption Scheme

In this chapter, the proposed scheme consists of two parts: compression encryption of the plain image and embedding of the encrypted data. The compression encryption of the plain image mainly includes chaos mapping and key allocation, DCT and adaptive data segmentation, and controllable integer compressive sensing. The embedding of encrypted data adopts the methods of replacing the LoSB data and match correction. The entire scheme is clearly presented in Figure 7, and the detailed steps are described as follows:

1. Key allocation and random sequence generation: The key size is 512 bits. Using the Equations (10) and (11), calculate $x_0, y_0, z_0, a, b, c, k, \gamma$ and N from the key, and input these parameters into the chaos mapping function to obtain random sequences sq_1 and sq_2 . Reverse the byte order of the key to obtain key' and repeat the same process to obtain random sequences sq_3 and sq_4 using key' .
2. Compression encryption of plain image: Divide the plain image into 8×8 blocks. Perform DCT, quantization, and DC component gradient processing on each block. Then, use adaptive data segmentation to separate each block into non-sparse part D_{ns} and sparse part D_s . Shuffle the sparse part using the sorting index sq_1 generated by the key, and evenly distribute it using a cyclic matrix formed by random sequence sq_2 to achieve controllable integer compressive sensing. Finally, encode the non-sparse

part and the compressed sparse part D_{s-cs} . Apply random bit flipping using random binary sequence sq_3 to the encoded data to obtain the encrypted data D_C .

3. Embedding of compressed encrypted data: Perform a modulo operation on the carrier image I_{Ca} according to the size of the compressed encrypted data of the plain image to obtain the carrier image I'_{Ca} and the LoSB data D_{CaL} . Extract a random sequence from sq_4 with the same length as D_{CaL} to obtain the sorting index sq_{i4} . Sort D_{CaL} using this index. Generate a blank matrix D_{CaL0} of the same size as D_{CaL} . Fill in the compressed encrypted data D_C sequentially and divide it into 2×2 blocks. Sort each block using the remaining random sequence from sq_4 and compare it with the LoSB data removed by the modulo operation. If the sorted data has a smaller sum of absolute deviations from the carrier image's LoSB, record it as 1; otherwise, record it as 0, and continue filling in D_{CaL0} until the match correction overlaps with the unfilled area. Fill the remaining area with the original LoSB data of the carrier image, and record the positions of various data. After filling the data, sort D_{CaL0} using the sorting index sq_{i4} in reverse order and embed it into the LoSB of I'_{Ca} .

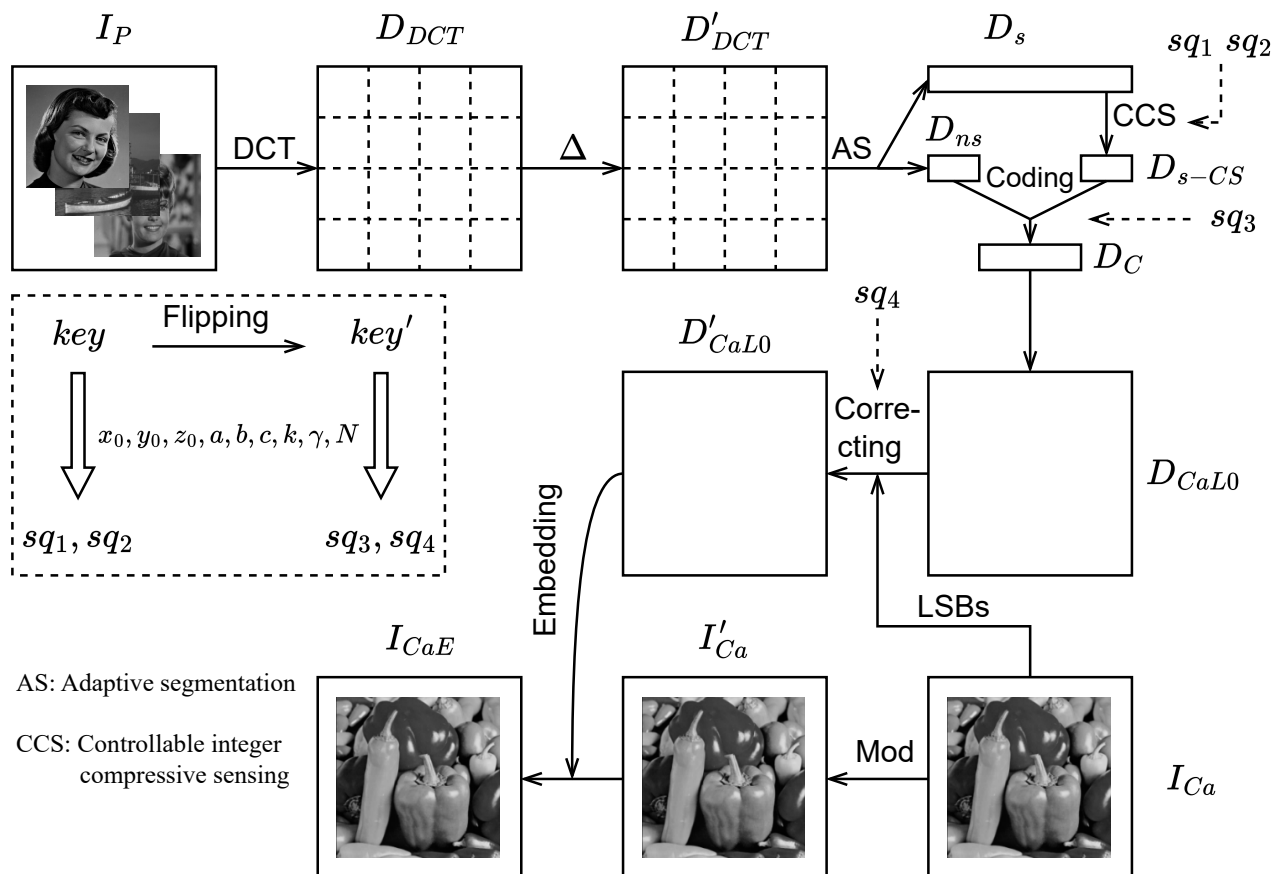


Figure 7. Flowchart of image encryption and hiding scheme based on controllable compressive sensing and adaptive matching embedding.

3. Simulation Experiment and Performance Analysis

In this section, we conducted a series of experimental analyses on the quality of the encrypted carrier image, the quality of the decrypted plain image, the capacity of the carrier, the security, and the efficiency of the scheme. All experiments were conducted using MATLAB 2018b (version 9.5, The MathWorks Inc., Natick, MA, USA) on a personal computer equipped with an Intel® Core™ i7-1260P CPU @ 2.1 GHz (Intel Corporation,

Santa Clara, CA, USA) and 16 GB of memory. The image-related experiments involved images with a size of 512×512 .

3.1. Image Quality and Carrier Capacity

To demonstrate the proposed scheme's capabilities in image compression, encryption, and hiding, we conducted experiments in which one or multiple plain images were compressed, encrypted, and hidden within a carrier image of the same size. It was ensured that the visual effect of the carrier image remained unchanged after embedding the encrypted information. This section presents the results of compressing, encrypting, and hiding 1, 2, 3, and 6 plain images in the carrier image shown in Figure 8, as shown in Figure 9. To validate the stability and practicality of the proposed scheme, the Peak Signal-to-Noise Ratio (PSNR) of the decrypted plain image, the PSNR of the encrypted carrier image, and the Mean Structural Similarity (MSSIM) of the encrypted carrier image after extracting the embedded information were calculated. $PSNR_d$, $PSNR_c$, and $MSSIM_c$ represent the PSNR of the decrypted plain image, PSNR of the encrypted carrier image, and MSSIM of the encrypted carrier image, respectively. Table 2 presents the PSNR and MSSIM results of the scheme when different numbers of plain images were embedded in the same carrier image. Table 3 presents the PSNR and MSSIM results of the scheme when different carrier images were used. From these experiments, it is observed that the proposed scheme can embed a greater number of images while maintaining the quality of the carrier image. Combining the results from Figure 9 and Table 2, it can be seen that after embedding four images, the PSNR of the carrier image is 38.82, and there is no significant visual difference from the original image. The average value of the decrypted images is 34.50. Even when embedding six plain images with a compression quality parameter of 30, there is a slight decrease in the visual quality of the carrier image, but no significant anomalies. The average PSNR of the decrypted plain images and the encrypted carrier images are 31.95 and 32.18, respectively. From the results in Table 3, it can be inferred that using different carrier images has a minimal impact on the quality of the carrier images.

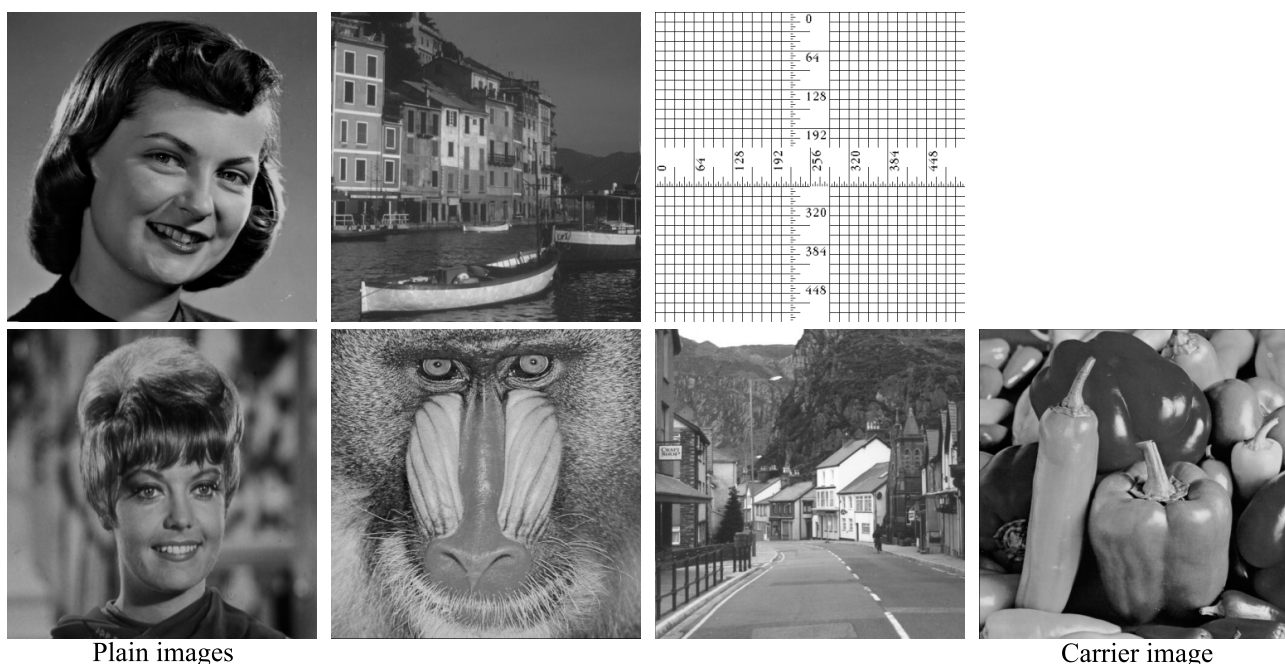


Figure 8. The plain images and carrier images.



$Q_v = 90$, PSNR = 41.86



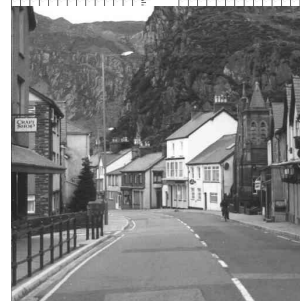
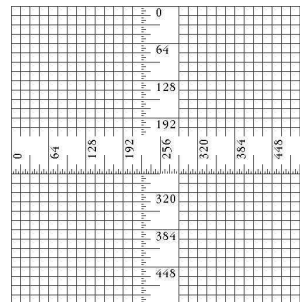
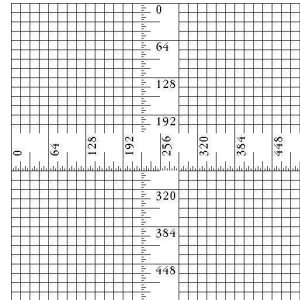
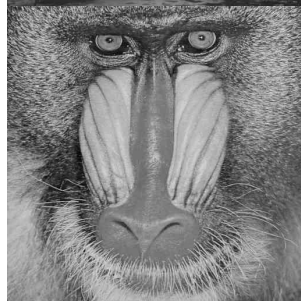
$Q_v = 80$, PSNR = 39.35/36.39



$Q_v = 50$, PSNR = 36.55/34.25/32.30



$Q_v = 30$, PSNR = 34.98/32.90/28.47/36.27/26.44/32.65



PSNR = 44.97



PSNR = 39.66



PSNR = 39.31



PSNR = 32.19

Figure 9. Decrypted images and encrypted carrier images when embedding different numbers of plain images in a single carrier image.

Table 2. Experimental results of encrypting and hiding different numbers of plain images.

Plain Image	Carrier Image	qv_d	$\overline{\text{PSNR}}_d$	PSNR_c	MSSIM_c
Girlface	Peppers	70	38.06	51.56	0.9997
		90	41.84	45.00	0.9985
		95	44.77	39.08	0.9936
Girlface Portofino	Peppers	65	37.59 35.08	45.22	0.9985
		80	39.37 36.39	39.62	0.9934
Girlface Portofino Ruler	Peppers	50	36.54 34.25 32.30	39.28	0.9933
Girlface Portofino Ruler Zelda	Peppers	40	35.73 33.66 31.50 37.10	38.82	0.9933
Girlface Portofino Ruler Zelda Baboon Town	Peppers	30	34.99 32.90 28.47 36.27 26.44 32.65	32.18	0.9730

Table 3. Experimental results of embedding different numbers of plain images in different carrier images.

Plain Image	Carrier Image	qv_d	$\overline{\text{PSNR}}_d$	PSNR_c	MSSIM_c
Group 1	Peppers	90	41.84	45.00	0.9985
	Elaine			44.94	0.9985
	Sailboat			44.97	0.9986
Group 2	Peppers	65	36.34	45.22	0.9983
	Elaine			45.20	0.9986
	Sailboat			45.23	0.9986
Group 3	Peppers	50	34.36	39.28	0.9933
	Elaine			39.23	0.9942
	Sailboat			39.24	0.9942
Group 4	Peppers	40	34.50	38.82	0.9933
	Elaine			38.77	0.9941
	Sailboat			38.80	0.9942
Group 5	Peppers	30	31.95	32.18	0.9730
	Elaine			32.25	0.9773
	Sailboat			33.19	0.9745

Group 1: Girlface. Group 2: Girlface, Portofino. Group 3: Girlface, Portofino, Ruler. Group 4: Girlface, Portofino, Ruler, Zelda. Group 5: Girlface, Portofino, Ruler, Zelda, Baboon, Town.

There is only a few published articles that focus on the encryption and hiding of multiple plain images in a single carrier image of the same size, and most of them achieve embedding of only two images into one carrier image. Here, we compare our proposed scheme with those studies [14,20] in terms of dual-image encryption and hiding, and we also conduct experiments to compare our scheme with multiple existing schemes in single-image encryption and hiding. For the single-image comparison in Table 4, to ensure a fair basis for evaluating carrier image quality, we controlled the experimental conditions such that the PSNR difference of the decrypted images across all compared schemes was

within 1%. Under this constraint, the carrier image quality ($PSNR_c$ and $MSSIM_c$) was then measured and compared. This design reflects the practical consideration that, given a fixed embedding capacity, a scheme that achieves higher carrier image quality at the same decryption fidelity is objectively superior. From Tables 5 and 4, it can be observed that our proposed scheme achieves favorable performance in both single-image and dual-image encryption and hiding. In the dual-image scenario (Table 5), our scheme simultaneously achieves higher $PSNR_d$, $PSNR_c$, and $MSSIM_c$ compared to Refs. [14,20], indicating that both the decrypted image quality and the carrier image fidelity are improved without a trade-off. In the single-image scenario (Table 4), the comparison was conducted under a controlled condition in which the PSNR difference between the decrypted images across all schemes was kept within 1%, ensuring a consistent baseline for evaluating carrier image quality. Under this constraint, our scheme achieves a mean $PSNR_c$ of 45.13 dB, compared to 35.21 dB and 34.59 dB for the compared methods.

Table 4. Comparison results of single-image encryption and hiding.

Plain Image	Carrier Image	Evaluating Index	Ref. [39]	Ref. [14]	Our Scheme
Girlface	Peppers	PSNR _c	32.35	36.11	45.00
		MSSIM _c	0.9257	0.9921	0.9985
Airplane	Baboon	PSNR _c	37.11	32.98	45.09
		MSSIM _c	0.9833	0.9951	0.9993
Girl	Godhill	PSNR _c	36.11	35.08	45.63
		MSSIM _c	0.9666	0.9966	0.9989
Barbara	Bridge	PSNR _c	35.56	34.17	44.81
		MSSIM _c	0.9583	0.9971	0.9981
Mean		PSNR _c	35.21	34.59	45.13
		MSSIM _c	0.9610	0.9952	0.9987

Bold values indicate the best result in each row. The comparison was conducted with the PSNR of decrypted images controlled within a 1% difference across all schemes.

Table 5. Comparison experiment results of dual-image encryption and hiding.

Plain Image	Carrier Image	PSNR _d	Ref. [14]		PSNR _d	Our Scheme	
			PSNR _c	MSSIM _c		PSNR _c	MSSIM _c
Zelda Woman	Sailboat	33.61 34.57	34.79	0.9952	38.67 41.00	44.96	0.9986
Lena Girl	Peppers	32.15 32.96	34.90	0.9946	36.83 37.56	45.75	0.9983
Kidney Brain	Xray	33.61 34.57	34.79	0.9952	42.03 40.20	45.46	0.9978
Mean		33.58	34.83	0.9950	39.38	45.39	0.9982
* Mean of Ref. [20]					37.14	38.76	\

Bold values indicate the best result in each row. * The data is the experimental results using different images in Ref. [20].

Based on the experimental results shown in Figure 9 and Table 3, our scheme can hide up to six plain images. Under the tested configuration (compression quality parameter of 30), both the encrypted carrier image and the decrypted plain images exhibit acceptable visual quality. It should be noted that as the number of hidden images increases, a gradual degradation in the visual quality of the carrier image is observed, which represents an inherent trade-off between hiding capacity and image fidelity. These experimental data

demonstrate that our proposed scheme can embed a much larger number of plain images compared to existing algorithms while maintaining the same quality of the decrypted image. Additionally, in the stability testing, we conducted experiments using a large number of personal photos (100 images) as plain images and web gallery images (20 images) as carrier images. Partial images used in this experiment are shown in Figure 10. We performed 100 random experiments of image encryption and hiding by selecting 1 to 6 plain images and 1 carrier image each time. The average PSNR, standard deviation, and confidence intervals of the decrypted images and carrier images were recorded, as shown in Table 6. Since the plain images were random snapshots, their complexity varied. Our scheme is based on JPEG compression, and for complex images like P1 and P12 in Figure 10, the compression capability is relatively lower, resulting in larger standard deviations in the PSNR values of the decrypted images. However, the standard deviations in the PSNR values of the carrier images were relatively small, indicating that the quality of the carrier image is less affected by the complexity of the plain images, and the impact becomes even smaller as the number of embedded images increases.



Figure 10. Plain images and carrier images in stability testing.

Table 6. Stability testing experiment of our scheme (100 random trials).

No. of Images	PSNR _d (dB)			PSNR _c (dB)		
	Mean $\pm$ SD	95% CI	Range	Mean $\pm$ SD	95% CI	Range
1	42.51 $\pm$ 3.40	[41.84, 43.17]	[36.2, 49.1]	44.90 $\pm$ 1.23	[44.66, 45.14]	[42.2, 47.6]
3	35.28 $\pm$ 3.63	[34.57, 36.00]	[28.4, 42.3]	39.22 $\pm$ 0.96	[39.03, 39.41]	[37.3, 41.1]
6	33.44 $\pm$ 4.13	[32.63, 34.26]	[25.9, 41.2]	32.54 $\pm$ 0.91	[32.36, 32.72]	[30.1, 34.6]

SD: standard deviation. CI: confidence interval of the mean. Range: [min, max] over 100 trials. The 95% confidence intervals were computed assuming approximate normality of the PSNR distribution, which was confirmed by the Shapiro-Wilk test ($p > 0.05$ for all cases).

Some schemes have different ways of embedding information, making it difficult to compare in terms of the number and quality of embedded images. On the other hand, many schemes are only applied to embedding a small number of bits. Here, we conducted experiments on the quality (PSNR) of the carrier images under different carrying conditions of our scheme and compared them with the specific carrying capacity of other schemes, including pairwise PEE-based methods [21,22], recursive code construction and wavelet-based approaches [24,25], and adaptive steganography using optimization techniques [26–28]. The results are shown in Figure 11. It is evident that our scheme produces higher-quality carrier images after encryption and hiding under the same carrying conditions than all methods except the one proposed in Reference [24]. The method in Reference [24] shows similar results to ours, but it requires a significant amount of computation [24]. In terms of efficiency, our scheme is more efficient than it.

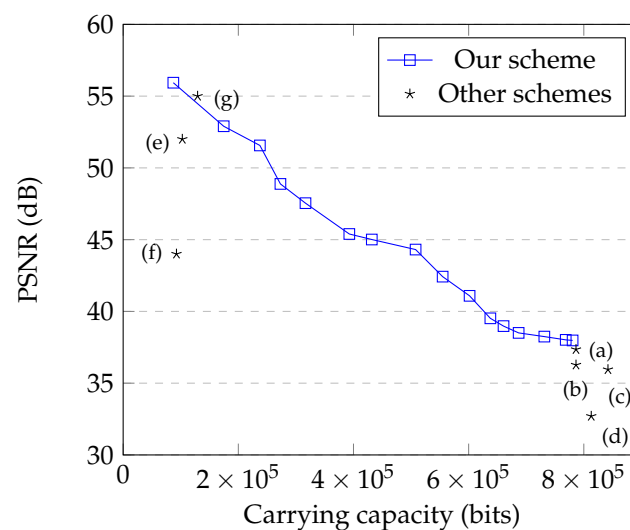


Figure 11. Comparison of carrying capacities in different schemes. (a) Hassaballah et al. (2021) [28]; (b) Muhuri et al. (2020) [27]; (c) Hameed et al. (2019) [26]; (d) Atta et al. (2018) [25]; (e) Dragoi et al. (2014) [22]; (f) Ou et al. (2013) [21]; (g) Hou et al. (2018) [24].

3.2. Security

In image encryption systems, security fundamentally relies on two critical factors: the robustness of key generation and distribution, and the statistical quality of the pseudorandom sequences used for encryption. A sufficiently large key space combined with high-quality, unpredictable randomness is essential to resist brute-force attacks, statistical analysis, and cryptanalytic reconstruction.

3.2.1. Practical Key Space Analysis

Based on the parameter ranges defined for the Fractal–Chaos Hybrid Map, the effective key space can be evaluated by considering both continuous and discrete parameters. The secret key consists of the initial conditions (x_0, y_0, z_0) , the control parameters (a, b, c, k) , the fractal perturbation strength γ , and the integer iteration bound N , whose admissible ranges are

$$x_0, y_0, z_0 \in [0, 1), \quad a \in [8, 18], \quad b \in [20, 40], \quad c \in [2, 4], \\ k \in [0, 1], \quad \gamma \in [0, 0.2], \quad N \in \{20, 21, \dots, 83\}.$$

For practical digital implementations, continuous parameters are represented with finite precision. Assuming a conservative precision of 10^{-15} (approximately 2^{-50}) for each real-valued parameter, the number of distinguishable values is approximately

$$|x_0| = |y_0| = |z_0| \approx 10^{15}, \\ |a| \approx \frac{18-8}{10^{-15}} = 10^{16}, \quad |b| \approx \frac{40-20}{10^{-15}} = 2 \times 10^{16}, \\ |c| \approx \frac{4-2}{10^{-15}} = 2 \times 10^{15}, \quad |k| \approx \frac{1}{10^{-15}} = 10^{15}, \\ |\gamma| \approx \frac{0.2}{10^{-15}} = 2 \times 10^{14}.$$

The discrete parameter N contributes

$$|N| = 83 - 20 + 1 = 64$$

possible values.

Therefore, the total practical key space $\mathcal{K}_{\text{eff}}$ can be approximated as

$$\mathcal{K}_{\text{eff}} \approx (10^{15})^3 \times 10^{16} \times 2 \times 10^{16} \times 2 \times 10^{15} \times 10^{15} \times 2 \times 10^{14} \times 64 \approx 5.12 \times 10^{123}. \quad (20)$$

Expressed in binary form, this corresponds to

$$\log_2 \mathcal{K}_{\text{eff}} \approx 411 \text{ bits}.$$

This key space is far larger than the commonly accepted security threshold of 2^{100} and provides a substantial margin against brute-force attacks. A comparison with existing schemes in terms of key space is shown in Table 7, where the proposed scheme has a larger key space and the key in the proposed scheme is correlated with the plaintext, making full use of every bit of the key. Moreover, due to the extreme sensitivity of the hybrid map to both initial conditions and parameter variations, many parameter combinations that differ by less than machine precision still produce completely different chaotic trajectories. Consequently, the effective key space of the proposed scheme is sufficiently large for practical cryptographic applications and provides strong resistance against exhaustive key-search attacks.

Table 7. Comparison of key space.

Scheme	Ref. [39]	Ref. [14]	Our Scheme
Key space	$2^{249.14}$	2^{376}	2^{411}

For the sensitivity analysis of the key, we conducted decryption result tests by sequentially modifying 1 key bit and randomly selecting two bits for 512 tests. We used the ‘Girlface’ and ‘Girl’ images as plain images, and ‘Peppers’ as the carrier image. The

compression quality parameter for the plain images was set to 65. The test results are shown in Table 8. Here, $rate_c$ represents the change rate of random binary sequences under different keys, and a value close to 0.5 indicates that the random sequences generated by different keys are almost uncorrelated. $rate_{c-min}$, $rate_{c-max}$, and $rate_{c-avg}$ represent the minimum, maximum, and average values of all tests, respectively. The calculation method is shown in Equation (21). P_{SD} represents the probability of successfully obtaining the decrypted image. The experimental results reflect that the proposed scheme is highly sensitive to the key and demonstrate the sufficient security of the scheme.

$$rate_{change} = \frac{\text{diff}(\text{FCHM}(k) > 0.5, \text{FCHM}(k') > 0.5)}{N} \quad (21)$$

Here, diff calculates the number of different bits between two binary sequences, $\text{FCHM}(k)$ represents the use of key k to generate the parameters for the sequence, and N is the number of $\text{FCHM}(k)$.

Table 8. Key sensitivity experimental results.

Key	$rate_{c-min}$	$rate_{c-max}$	$rate_{c-avg}$	P_{SD}
1-bit change	0.4996	0.5003	0.4999	0
2-bit change	0.4997	0.5003	0.5000	0

3.2.2. Statistical Randomness Evaluation

To quantify the contribution of the Mandelbrot perturbation to statistical randomness, we conduct an ablation study comparing the pure Lorenz map ($\gamma = 0$) with the proposed hybrid map at three representative iteration limits within the adopted parameter range ($\gamma = 0.05$, $N = 10, 20, 40, 83, 200$). For each configuration, 100 independent sequences of 10^6 values are generated with randomly chosen keys and evaluated using the NIST SP 800-22 test suite (16 tests, $\alpha = 0.01$). A sequence is considered to pass if no more than 2 out of 16 tests are failed.

The results in Table 9 reveal two important findings. First, the pure Lorenz map achieves a NIST pass rate of only 0.88, indicating that approximately 12% of its output sequences exhibit detectable statistical non-randomness—an unacceptable rate for cryptographic applications. The Mandelbrot perturbation systematically eliminates these defects: the pass rate rises to 0.98 at $N = 20$ and reaches 1.00 at $N = 40$, confirming that the fractal component provides a measurable and essential improvement in statistical quality.

Table 9. Ablation study of the Mandelbrot fractal component.

Configuration	Time (s)	Overhead	Security Metrics	
			NIST Pass Rate	Key Sensitivity
Pure Lorenz ($\gamma = 0$)	0.65	—	0.88	0.497
Hybrid ($N = 10$)	1.54	+137%	0.95	0.501
Hybrid ($N = 20$)	1.77	+173%	0.98	0.499
Hybrid ($N = 40$)	2.15	+231%	1.00	0.500
Hybrid ($N = 83$)	2.90	+346%	1.00	0.501
Hybrid ($N = 200$)	4.89	+655%	1.00	0.501

Overhead: relative to Pure Lorenz. NIST pass rate: proportion of 100 sequences with ≤ 2 failed tests out of 16 ($\alpha = 0.01$).

Second, from a computational perspective, the overhead relative to the pure Lorenz baseline ranges from +173% ($N = 20$) to +346% ($N = 83$, the upper bound of the adopted parameter range). At $N = 20$, the NIST pass rate already reaches 0.98, providing a substantial security improvement at moderate cost. At $N = 83$, the overhead of +346%

remains within an acceptable range for practical deployment, while the pass rate is maintained at 1.00. While this represents a non-trivial increase in the sequence generation micro-benchmark, it is important to contextualize this cost within the complete encryption pipeline. Sequence generation accounts for only a small fraction of the total encryption time; the dominant operations (DCT transform, compressive sensing, and data embedding) are independent of the chaotic map. Consequently, the end-to-end throughput impact is substantially smaller than Table 9 suggests in isolation. These results confirm that the adopted parameter range $N \in [20, 83]$ strikes a favorable balance between statistical security and computational efficiency.

We evaluated the randomness of the proposed FCHM keystream using the NIST SP 800-22 statistical test suite (National Institute of Standards and Technology, Gaithersburg, MD, USA) [40] on 200 independent sequences of 10^6 bits ($\alpha = 0.01$). Three comparison methods (CSPRNG(SHA-256)-based, 2P-ELCM-based [30], hyperchaos-based [36]) were tested under identical conditions. A test passes if the proportion of successful sequences falls within $[0.9689, 1.0111]$ and the uniformity p -value ≥ 0.0001 . Results are summarized in Table 10.

Table 10. Summary of NIST SP 800-22 batch test results (200 sequences, 10^6 bits per sequence, $\alpha = 0.01$). “PR” = pass rate; “Unif” = uniformity of p -values.

Test Name	Our FCHM		CSPRNG		2P-ELCM		4D Hyperchaos	
	PR	Unif	PR	Unif	PR	Unif	PR	Unif
01. Frequency (Monobit)	1.000	PASS	0.985	PASS	0.525	FAIL	0.980	PASS
02. Block Frequency	0.990	PASS	0.975	PASS	0.775	FAIL	0.975	PASS
03. Runs	0.985	PASS	0.990	PASS	0.415	FAIL	0.980	PASS
04. Longest Run of Ones	0.995	PASS	0.990	PASS	0.140	FAIL	0.990	PASS
05. Binary Matrix Rank	0.980	PASS	0.985	PASS	0.965	FAIL	0.995	PASS
06. Spectral (DFT)	0.990	PASS	0.990	PASS	0.015	FAIL	0.990	PASS
07. Non-overlapping Template	0.995	PASS	0.985	PASS	0.650	FAIL	0.995	PASS
08. Overlapping Template	0.985	PASS	0.970	PASS	0.290	FAIL	0.990	PASS
09. Universal Statistical	0.995	PASS	0.980	PASS	0.675	FAIL	0.990	PASS
10. Linear Complexity	0.995	PASS	0.975	PASS	0.935	FAIL	0.995	PASS
11. Serial	0.990	PASS	0.985	FAIL *	0.000	FAIL	0.965	FAIL **
12. Approximate Entropy	0.990	PASS	0.985	PASS	0.000	FAIL	0.995	PASS
13. Cumulative Sums (Forward)	1.000	PASS	0.990	PASS	0.545	FAIL	0.995	PASS
14. Cumulative Sums (Backward)	1.000	PASS	0.990	PASS	0.520	FAIL	0.995	PASS
15. Random Excursions	0.940	FAIL †	0.930	FAIL †	0.905	FAIL †	0.860	FAIL †
16. Random Excursions Variant	0.910	FAIL †	0.905	FAIL †	0.885	FAIL †	0.895	FAIL †

Bold values indicate the best pass rate among all schemes for each test. * Uniformity p -value = 0.000043 < 0.0001.

** Pass rate = 0.965 < 0.9689. † Both pass rate and uniformity fail.

The proposed FCHM passes 14 out of 16 tests, failing only the two excursion-based tests—a common limitation for many chaotic RNGs. It achieves perfect or near-perfect pass rates (0.980–1.000) across all other tests, with uniform p -value distributions. In contrast, 2P-ELCM fails most tests with pass rates ranging from 0.000 to 0.775. CSPRNG fails the Serial uniformity test, and Hyperchaos fails the Serial proportion test while exhibiting a significantly worse Random Excursions pass rate (0.860 vs. 0.940 for our method). These results demonstrate that the proposed FCHM provides superior statistical randomness, making it well-suited for secure image encryption. In terms of generation throughput, FCHM achieves 1.993 Mbps, which is markedly higher than 2P-ELCM (1.062 Mbps) and the 4D hyperchaotic system (0.067 Mbps). We note that the CSPRNG-based generator achieves a higher throughput of 12.377 Mbps; however, this advantage stems from deeply optimized implementations that leverage CPU instruction-level acceleration and dedicated hardware support and does not

reflect inherent algorithmic superiority. For the chaotic generators compared, FCHM offers the best balance between statistical quality and computational efficiency.

Building on the statistical randomness of the FCHM keystream demonstrated above, we now evaluate the security of the complete encryption pipeline through standard image-specific metrics.

3.2.3. Statistical and Differential Attack Resistance

It should be noted that the proposed scheme embeds the encrypted information into a carrier image for transmission. To isolate and evaluate the security of the encryption component, the following analysis is conducted on the compressed and encrypted data prior to the embedding stage.

Figure 12 presents a comprehensive visual security evaluation using the “Camera” image as a representative example. Figure 12a shows the original plaintext image, while Figure 12b displays the encrypted data, which exhibits a noise-like appearance with no discernible structural information. The histogram of the encrypted data, shown in Figure 12c, is uniformly distributed across all intensity levels, confirming that the ciphertext provides no statistical leakage about the plaintext distribution. Furthermore, the adjacent pixel correlation scatter plots in the horizontal, vertical, and diagonal directions (Figure 12d) show points uniformly distributed across the entire coordinate plane, indicating that pixel correlations have been effectively destroyed.

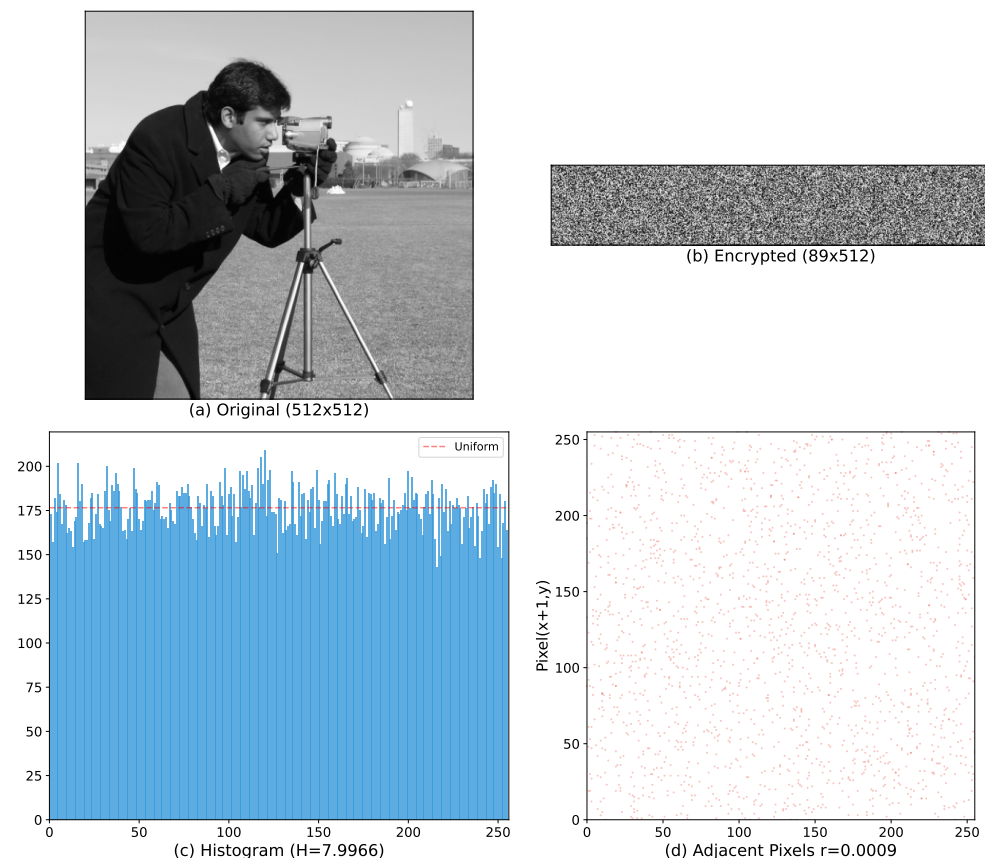


Figure 12. Visual security evaluation: (a) plaintext image, (b) encrypted image, (c) histogram of encrypted image, and (d) adjacent pixel correlation scatter plots.

Table 11 reports the quantitative security metrics for three test images of different types. The information entropy of all encrypted images exceeds 7.999, extremely close to the theoretical maximum of 8 bits per pixel, confirming that the ciphertext is statistically indistinguishable from random data. The horizontal, vertical, and diagonal correlation

coefficients are all within ± 0.05 of zero (maximum absolute value 0.0349), indicating that adjacent pixel correlations have been effectively eliminated. In terms of differential attack resistance, all NPCR values exceed the strict threshold of $\geq 99.61\%$ (ranging from 99.6467% to 99.6985%), and the UACI values are consistently close to the ideal 33.46% (ranging from 33.3684% to 33.4621%), demonstrating that a single-pixel change in the plaintext propagates to nearly all ciphertext pixels. Additionally, the chi-square test yields p -values well above the 0.01 significance level for all images, formally confirming the uniformity of the ciphertext histograms and corroborating the visual histogram analysis.

Table 11. Security metrics of encrypted images.

Image	Entropy	Corr H	Corr V	Corr D	NPCR (%)	UACI (%)	χ^2	p -Value
Camera	7.9992	−0.0196	0.0146	0.0349	99.6985	33.3684	237.6	0.7764
Brick	7.9990	−0.0266	0.0219	−0.0243	99.6467	33.4621	228.7	0.8806
Astronaut	7.9991	−0.0124	0.0148	−0.0068	99.6603	33.3860	265.4	0.3144
Ideal	8.0000	≈ 0	≈ 0	≈ 0	≥ 99.61	≈ 33.46	≈ 255	≥ 0.01

3.2.4. Resistance to Known-Plaintext and Chosen-Plaintext Attacks

The proposed scheme resists known-plaintext and chosen-plaintext attacks through multiple complementary mechanisms. First, the secret key is derived from the SHA-512 hash of the plain image combined with random noise, ensuring that different plaintexts—even those differing by a single pixel—produce entirely different keystreams. Consequently, known plaintext–ciphertext pairs for one image are useless for attacking another, and chosen-plaintext attacks are thwarted because the attacker cannot predict the random noise component used in key generation. Second, even if a fixed master key were used, the effective key space of approximately 2^{411} makes brute-force enumeration infeasible, and the Fractal–Chaos Hybrid Map’s extreme key sensitivity combined with the non-algebraic nature of the Mandelbrot perturbation prevents attackers from reconstructing the internal state or key parameters from observed keystreams. Finally, the embedding of encrypted data into a carrier image adds an additional layer of obfuscation: an attacker must first correctly extract the encrypted payload before attempting any cryptanalytic attack, further increasing the attack complexity.

3.3. Efficiency

In addition to the security analysis, this section also compares the proposed scheme with existing excellent schemes in terms of encryption and decryption efficiency. To conduct the experimental comparison, the proposed scheme in this paper only focuses on the encryption and decryption of a single plain image. The results are shown in Table 12. Furthermore, this section also conducts experimental analysis on the encryption and hiding of different numbers of plain images, as shown in Table 12.

Table 12. Encryption and decryption time of different schemes (unit: s).

Scheme	N_p	Encryption	Decryption
Ref. [39]	1	0.3100	2.9724
Ref. [14]	1	0.3660	2.7178
	2	0.7280	5.5053
Our Scheme	1	0.6941	1.7453
	2	1.0173	2.9736
	3	1.3857	3.8511
	6	2.4006	6.3487

N_p is the number of embedded plain images. Bold values indicate the best performance for the same N_p .

The encoding method, introduced to reduce the compression rate and increase the number of embedded images, and the matching correction method, introduced to improve image quality after data embedding, slightly reduce the encryption efficiency of the proposed scheme compared to the other two methods. These two methods increase the time consumption. However, in the decryption experiment, the proposed scheme outperforms the other two schemes. This result is attributed to the adaptive data segmentation and the controllable integer compressive sensing, both enabled by the cyclic matrix FFT acceleration used in the scheme.

4. Conclusions

In this paper, we propose a multi-image encryption and hiding scheme based on controllable compressive sensing and adaptive matching embedding. Based on image encryption, the encrypted information is embedded in a carrier image for transmission, making it difficult to identify the encrypted information and reducing the risk of attack to a certain extent. In terms of security, this paper proposes a more comprehensive key distribution method, providing an effective key space of approximately 2^{411} . The Fractal–Chaos Hybrid Map generates random sequences with greater efficiency and stronger security. In addition, the adaptive embedding method based on modular arithmetic and matching correction improves the quality of the carrier image while embedding the data, reducing visual artifacts and lowering the risk of attack. In terms of processing efficiency, the proposed scheme combines DCT with adaptive data segmentation and introduces controllable integer compressive sensing, which greatly improves the efficiency of encryption and decryption. In terms of information carrying capacity, encoding and controllable integer compressive sensing play a crucial role.

Simulation experiments and performance analysis show that the proposed scheme achieves excellent security, operational efficiency, and information-carrying capacity. It performs well in both single- and double-image encryption and hiding, and can encrypt and hide up to 6 plain images in a single carrier image of the same size. The decrypted plain images and the encrypted carrier image both meet the requirements of most applications in terms of quality. The key space and key sensitivity of the scheme meet the requirements for image encryption. The scheme also achieves good operational efficiency, with slightly lower encryption efficiency and significantly higher decryption efficiency compared to the reference schemes. Additionally, the stability of the scheme was tested, and it showed good stability when embedding images with random complexity.

To enhance the information-carrying capacity, the proposed scheme introduces encoding. To improve the quality of the carrier image and reduce visual artifacts, matching correction is introduced. The encoding and matching correction methods reduce the efficiency of encryption and decryption, and the high compression rate of the encoding method makes the scheme largely lose its fault tolerance. To evaluate residual robustness,

we conducted preliminary tests under salt-and-pepper noise ($d = 0.002, 0.005$), cropping (5%, 10%), and simulated packet loss (2%, 5%). As shown in Figure 13, the decrypted images retain recognizable content under moderate distortions, indicating that the scheme possesses a basic level of tolerance before complete failure. Nevertheless, for applications that demand strong robustness, a future direction is to balance compression and fault tolerance by sacrificing a reasonable amount of storage space, potentially through the integration of error-correcting codes prior to embedding. Furthermore, extending the proposed encryption and hiding framework to support RGB color images would enable direct comparison with a broader range of state-of-the-art color image hiding schemes, which is another important direction for future work.

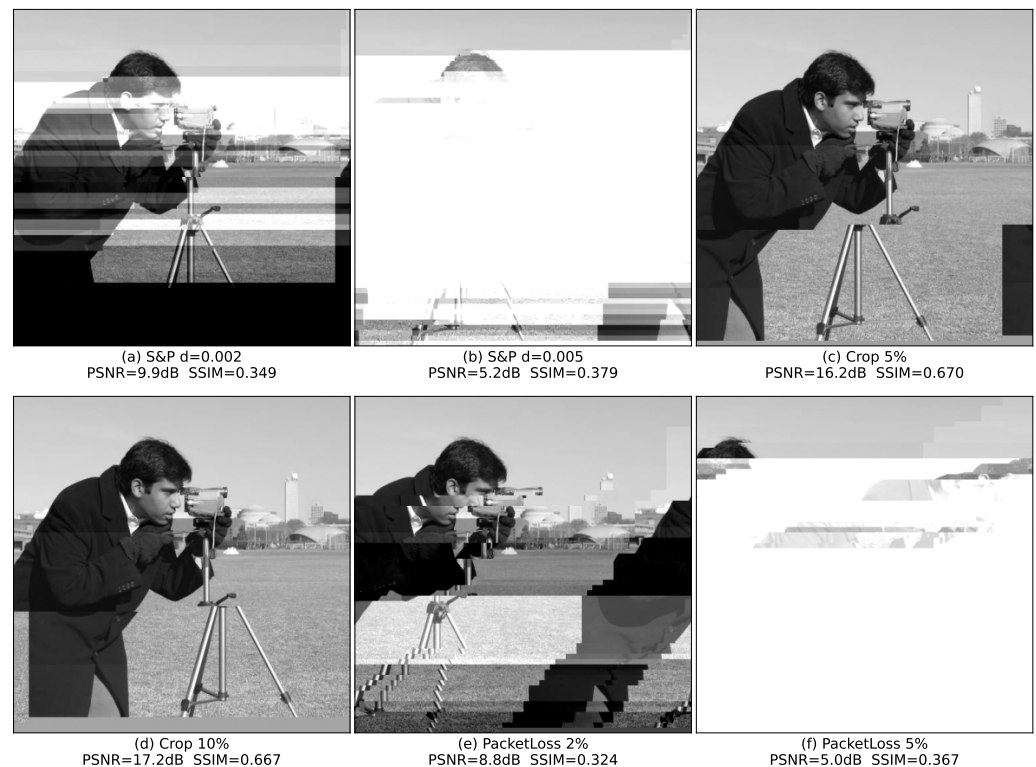


Figure 13. Robustness test results: decrypted images under (a) S&P noise ($d = 0.002$), (b) S&P noise ($d = 0.005$), (c) 5% crop, (d) 10% crop, (e) 2% packet loss, and (f) 5% packet loss.

Author Contributions: Z.L. and C.W. conceived and designed the experiments; C.W. performed the experiments and analyzed the data; W.C. and Y.L. contributed to data curation and software implementation; Z.L. supervised the project and acquired funding. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Fund of the National Key Research and Development Program of China (No. 2022YFD2401304); Southern Marine Science and Engineering Guangdong Laboratory (Zhanjiang) (No. ZJW-2023-04); the Scientific Research Project of the Education Department of Guangdong Province (Nos. 2024ZDZX4060, 2022GCZX001, 2023ZDZX4012); Special Fund Project for Talent Development Strategy of Guangdong Province (No. 2024R1003); and the Program for Scientific Research Start-up Funds of Guangdong Ocean University (No. 060302102404).

Data Availability Statement: The data and code that support the findings of this study are available from the corresponding author upon reasonable request, due to restrictions related to the ongoing research and the intellectual property involved in the encryption scheme.

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The

funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Liu, S.; Guo, C.; Sheridan, J.T. A review of optical image encryption techniques. *Opt. Laser Technol.* **2014**, *57*, 327–342.
2. Kaur, M.; Kumar, V. A comprehensive review on image encryption techniques. *Arch. Comput. Methods Eng.* **2020**, *27*, 15–43.
3. Yao, M.; Deng, H.; Chen, Z.; Zhang, P. Efficient and secure image compression and encryption based on compressive sensing and four-dimensional hyperchaotic system. *Inf. Sci.* **2025**, *719*, 122430. <https://doi.org/10.1016/J.INS.2025.122430>.
4. Moulin, P.; O’Sullivan, J.A. Information-theoretic analysis of information hiding. *IEEE Trans. Inf. Theory* **2003**, *49*, 563–593.
5. Jiao, S.; Zhou, C.; Shi, Y.; Zou, W.; Li, X. Review on optical image hiding and watermarking techniques. *Opt. Laser Technol.* **2019**, *109*, 370–380.
6. Jiang, X.; Xie, Y.; Zhang, Y.; Ye, Y.; Xu, F.; Li, L.; Su, Y.; Chen, Z. Reversible data hiding in encrypted images using reservoir computing based data fusion strategy. *IEEE Trans. Circuits Syst. Video Technol.* **2024**, *35*, 684–697.
7. Chan, C.K.; Cheng, L.M. Hiding data in images by simple LSB substitution. *Pattern Recognit.* **2004**, *37*, 469–474.
8. Chang, C.C.; Lin, C.C.; Tseng, C.S.; Tai, W.L. Reversible hiding in DCT-based compressed images. *Inf. Sci.* **2007**, *177*, 2768–2786.
9. Jia, Y.; Yin, Z.; Zhang, X.; Luo, Y. Reversible data hiding based on reducing invalid shifting of pixels in histogram shifting. *Signal Process.* **2019**, *163*, 238–246.
10. Panigrahi, R.; Padhy, N. An effective steganographic technique for hiding the image data using the LSB technique. *Cyber Secur. Appl.* **2025**, *3*, 100069. <https://doi.org/10.1016/J.CSA.2024.100069>.
11. Yin, Z.; Xiang, Y.; Zhang, X. Reversible data hiding in encrypted images based on multi-MSB prediction and Huffman coding. *IEEE Trans. Multimed.* **2019**, *22*, 874–884.
12. Sheidani, S.; Mahmoudi-Aznavah, A.; Eslami, Z. CPA-secure privacy-preserving reversible data hiding for JPEG images. *IEEE Trans. Inf. Forensics Secur.* **2021**, *16*, 3647–3661.
13. Luo, Y.; Qin, J.; Xiang, X.; Tan, Y.; Liu, Q.; Xiang, L. Coverless real-time image information hiding based on image block matching and dense convolutional network. *J. Real-Time Image Process.* **2020**, *17*, 125–135.
14. Wang, X.; Liu, C.; Jiang, D. An efficient double-image encryption and hiding algorithm using a newly designed chaotic system and parallel compressive sensing. *Inf. Sci.* **2022**, *610*, 300–325.
15. Hosny, K.M.; Magdi, A.; ElKomy, O.; Hamza, H.M. Digital image watermarking using deep learning: A survey. *Comput. Sci. Rev.* **2024**, *53*, 100662.
16. Bao, L.; Zhou, Y. Image encryption: Generating visually meaningful encrypted images. *Inf. Sci.* **2015**, *324*, 197–207.
17. Chai, X.; Wu, H.; Gan, Z.; Zhang, Y.; Chen, Y.; Nixon, K.W. An efficient visually meaningful image compression and encryption scheme based on compressive sensing and dynamic LSB embedding. *Opt. Lasers Eng.* **2020**, *124*, 105837.
18. Zhu, L.; Song, H.; Zhang, X.; Yan, M.; Zhang, T.; Wang, X.; Xu, J. A robust meaningful image encryption scheme based on block compressive sensing and SVD embedding. *Signal Process.* **2020**, *175*, 107629.
19. Wang, X.; Liu, C.; Jiang, D. A novel triple-image encryption and hiding algorithm based on chaos, compressive sensing and 3D DCT. *Inf. Sci.* **2021**, *574*, 505–527.
20. Guan, Z.; Jing, J.; Deng, X.; Xu, M.; Jiang, L.; Zhang, Z.; Li, Y. DeepMIH: Deep invertible network for multiple image hiding. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 372–390.
21. Ou, B.; Li, X.; Zhao, Y.; Ni, R.; Shi, Y.Q. Pairwise prediction-error expansion for efficient reversible data hiding. *IEEE Trans. Image Process.* **2013**, *22*, 5010–5021.
22. Dragoi, I.C.; Coltuc, D. Local-prediction-based difference expansion reversible watermarking. *IEEE Trans. Image Process.* **2014**, *23*, 1779–1790.
23. Gao, G.; Shi, Y.Q. Reversible data hiding using controlled contrast enhancement and integer wavelet transform. *IEEE Signal Process. Lett.* **2015**, *22*, 2078–2082.
24. Hou, D.; Zhang, W.; Yang, Y.; Yu, N. Reversible data hiding under inconsistent distortion metrics. *IEEE Trans. Image Process.* **2018**, *27*, 5087–5099.
25. Atta, R.; Ghanbari, M. A high payload steganography mechanism based on wavelet packet transformation and neutrosophic set. *J. Vis. Commun. Image Represent.* **2018**, *53*, 42–54.
26. Hameed, M.A.; Hassaballah, M.; Aly, S.; Awad, A.I. An adaptive image steganography method based on histogram of oriented gradient and PVD-LSB techniques. *IEEE Access* **2019**, *7*, 185189–185204.
27. Muhuri, P.K.; Ashraf, Z.; Goel, S. A novel image steganographic method based on integer wavelet transformation and particle swarm optimization. *Appl. Soft Comput.* **2020**, *92*, 106257.
28. Hassaballah, M.; Hameed, M.A.; Awad, A.I.; Muhammad, K. A novel image steganography method for industrial internet of things security. *IEEE Trans. Ind. Inform.* **2021**, *17*, 7743–7751.

29. Wang, C.; Song, L. An image encryption scheme based on chaotic system and compressed sensing for multiple application scenarios. *Inf. Sci.* **2023**, *642*, 119166.
30. Latoui, A.; Daachi, M.E.H. A two-parameter extended logistic chaotic map for modern image cryptosystems. *Digit. Signal Process.* **2024**, *148*, 104463.
31. Pandey, K.; Sharma, D. Novel image encryption algorithm utilizing hybrid chaotic maps and Elliptic Curve Cryptography with genetic algorithm. *J. Inf. Secur. Appl.* **2025**, *89*, 103995.
32. Xu, B.; Ye, X.; Wang, X.; Zhang, Z. Image encryption scheme based on compressed sensing and fractional quantum logistic-tent map. *Cybersecurity* **2025**, *8*, 56.
33. Mohamed, N.A.E.S.; Youssif, A.; El-Sayed, H.A.G. Fast and robust image encryption scheme based on quantum logistic map and hyperchaotic system. *Complexity* **2022**, *2022*, 3676265.
34. Zhou, Y.; Hua, Z.; Pun, C.M.; Chen, C.P. Cascade chaotic system with applications. *IEEE Trans. Cybern.* **2014**, *45*, 2001–2012.
35. Jia, M.; Jiang, H.; Li, W. Generation and application of novel Chua multi-scroll chaotic attractors. *Acta Phys. Sin.* **2019**, *68*, 61–73.
36. Chen, Z.; Chen, Z.; Long, B.; Liu, T.; Yao, M. A cross-channel color image encryption scheme based on a novel 4D hyperchaotic system and 3-layer Peano curve. *Expert Syst. Appl.* **2026**, p. 131145.
37. Yin, W.; Morgan, S.; Yang, J.; Zhang, Y. Practical compressive sensing with Toeplitz and circulant matrices. In *Proceedings of the Visual Communications and Image Processing 2010, Huangshan, China, 11–14 July 2010*; International Society for Optics and Photonics: Bellingham, WA, USA, 2010; Volume 7744, p. 77440K.
38. Pati, Y.C.; Rezaiifar, R.; Krishnaprasad, P.S. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Proceedings of the 27th Asilomar Conference on Signals, Systems and Computers*; IEEE: New York, NY, USA, 1993; pp. 40–44.
39. Wang, H.; Xiao, D.; Li, M.; Xiang, Y.; Li, X. A visually secure image encryption scheme based on parallel compressive sensing. *Signal Process.* **2019**, *155*, 218–232.
40. Barker, E.B.; Kelsey, J.M. *Recommendation for Random Number Generation Using Deterministic Random Bit Generators*; Technical Report NIST SP 800-90A Rev. 1; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2015. <https://doi.org/10.6028/NIST.SP.800-90Ar1>.

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.