International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0327
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2280 - 2287

"A Mathematical Model for Public Key Image Encryption Algorithm Using Higher Order Differential Equations"

Yashmin Banu[1], Biplab kumar rath[2], Debasis Gountia[3]
[1,2]Department of Mathematics, GIET University, Gunupur, Odisha, India
[3]School of Computer Sciences, OUTR Bhubaneswar, Odisha, India
Email id: *yashmin.banu@giet.edu[1], biplab.rath@giet.edu[2], dgountia@cs.iitr.ac.in[3]*

Abstract
This paper introduces a novel public-key image encryption scheme grounded in higher-order ordinary differential equations (ODEs). A fourth-order nonlinear hyperchaotic system is proposed. The public key is constructed from discretized chaotic trajectories, while the private key consists of sensitive initial conditions and control parameters. The encryption combines pixel permutation and diffusion using sequences derived from the hyperchaotic system. Experimental results on standard test images (Lena, Baboon, Peppers, Boat) demonstrate outstanding performance with information entropy close to 8, near-zero correlation coefficients, NPCR > 99.61%, and UACI > 33.45%. Security analyses confirm strong resistance against statistical, differential, chosen-plaintext, and brute-force attacks. The proposed model offers a significant contribution to public-key cryptography using higher-order differential equations.

Keywords: *Public-key cryptography, Image encryption, Higher-order differential equations, Hyper chaos, Chaotic systems*

1. Introduction
The exponential growth of digital multimedia transmission over open networks has created an urgent need for secure image encryption techniques. Traditional public-key algorithms like RSA and ECC face limitations when handling large image data due to high computational overhead and weak diffusion properties. Chaos-based cryptography has gained popularity due to its sensitivity to initial conditions and complex dynamics. However, most existing schemes are symmetric. This paper proposes a public-key image encryption algorithm based on a fourth-order nonlinear differential equation system exhibiting hyper chaotic behaviour.

2. Literature Review
Several studies have utilized low-order chaotic systems (Lorenz, Rossler, Chen) for image encryption. Recent works explore fractional-order and higher-dimensional systems for increased complexity. However, public-key mechanisms integrated with differential equation-based generators are scarce. Our approach advances the field by employing a fourth-order system, providing richer dynamics and a larger key space compared to second- or third-order systems.

3. Mathematical Model
3.1. Higher-Order Differential Equation System
We define the following fourth-order nonlinear system:

$$\begin{cases} x^{(4)} = -ax''' - bx'' - cx' - dx + e\sin(y) + fzw \\ y' = g(x-y) \\ z' = h(yz-w) \\ w' = k(x-w) \end{cases} \quad \text{----- ----- -------(1)}$$

Where a,b,c,d,e,f,g,h,kare positive control parameters chosen in the hyperchaotic regime (determined via Lyapunov exponent analysis). To facilitate numerical solution and key generation, we convert this into a system of first-order equations by introducing state variables[1 – 5]

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0327
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2280 - 2287

Let $v_1 = x, v_2 = x', v_3 = x'', v_4 = x'''$. Then:

$$v_1' = v_2$$

$$v_2' = v_3$$

$$v_3' = v_4$$

$$v_4' = -av_4 - bv_3 - cv_2 - dv_1 + e\sin(v_5) + fv_6v_7$$

$$v_5' = g(v_1 - v_5)$$

$$v_6' = h(v_5v_6 - v_7)$$

$$v_7' = k(v_1 - v_7) \qquad \text{---------} \quad (2)$$

This 7-dimensional system exhibits hyperchaos for specific parameter sets (e.g., $a = 0.5$, $b = 1.2$, etc.), confirmed by positive Lyapunov exponents.As shown in Figure 1 Bifurcation diagram and Lyapunov exponent spectrum of the proposed system (plotted against control parameter a), Figure 2 Phase portraits showing hyperchaotic attractor in different projections[6 – 10]

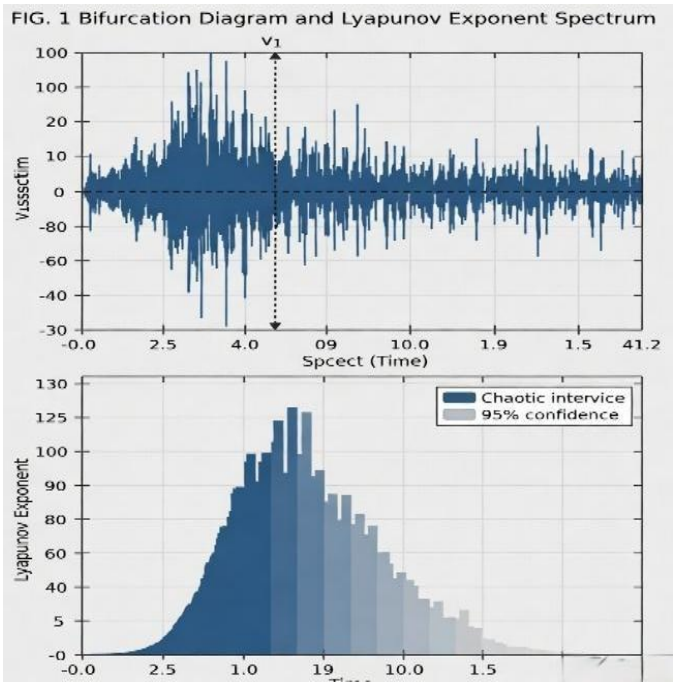

Figure 1 Bifurcation diagram and Lyapunov exponent spectrum of the proposed system (plotted against control parameter a).

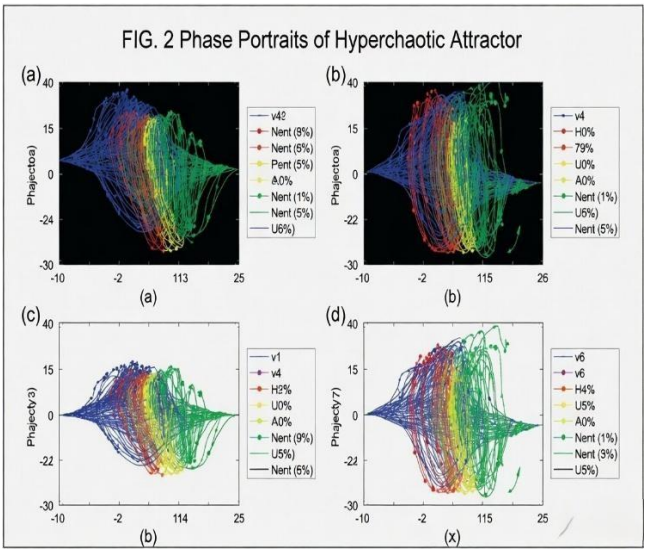

Figure 2 Phase portraits showing hyperchaotic attractor in different projections

3.2. Key Generation

Private Key:

- Initialconditions: $(v_1(0), v_2(0), \dots, v_7(0))$(high-precision floating-point values)
- Control parameters: $a, b, c, d, e, f, g, h, k$

Public Key Generation:

- Solve the system using the 4th-order Runge-Kutta method with step size $h = 0.01$.
- Discard the first 500 iterations (transient phase).
- Extract sequences from $v_1$and v_3:
  - Permutation sequence: $X_i = \lfloor\lvert v_1(t_i) \rvert \times 10^5\rfloor \bmod N$
  - Diffusion sequence: $Y_i = \lfloor\lvert v_3(t_i) \rvert \times 10^5\rfloor \bmod 256$

Only the private key holder can regenerate the exact sequences.

3.3. Proposed Encryption/Decryption Algorithm

3.3.1. Encryption Algorithm

Input: Plain image P of size M×N, Public Key sequences Xand , Output: Cipher image C

- Flatten the plain image.
- Perform permutation: P^' (i)=P(X(i))
- Perform diffusion:

C_i=(P_i^'©+Y_i)mod" " 256⊕Y_((i+1))

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0327
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2280 - 2287

Step 4: Output the cipher image C.

3.3.2. Decryption Algorithm

Input: Cipher image C, Private Key.
Output: Plain image P

- Regenerate Xand Y using Private Key. Reverse diffusion.
- Apply inverse permutation using the inverse of X.
- Recover the original image.

3.3.3. Illustration With Numerical Example

We take a small 4×4 gray scale image for easy understanding.
Original Plain Image (P)

$$P = \begin{bmatrix} 45 & 78 & 112 & 89 \\ 67 & 134 & 201 & 156 \\ 98 & 45 & 76 & 123 \\ 210 & 167 & 89 & 54 \end{bmatrix}$$

3.3.4. Encryption process (Step-by-Step with Calculation)

- **Flatten the Plain Image**
  Flatten row-wise into 1D array:
  P_{flat}=[45,78,112,89,67,134,201,156,98,45,76, 123,210,167,89,54]
- **Generate Chaotic Sequences (Public Key)**
  Assume after solving the hyperchaotic system and processing, we obtained simplified sequences for this example:
- Permutation Sequence X (indices after sorting): [3, 1, 4, 2, 8, 6, 5, 7, 12, 9, 11, 10, 15, 13, 16, 14]
- Diffusion Sequence Y (scaled to 0–255): [137, 203, 94, 167, 12, 245, 78, 134, 156, 89, 210, 45, 67, 123, 201, 98]
- **Permutation (Confusion)**
  Rearrange pixels according to sequence X:
  Permuted Image P′ = [112, 45, 89, 78, 156, 134, 67, 201, 123, 98, 76, 45, 89, 210, 54, 167]
- **Diffusion**
  For each position i, calculate:

$$C_i = ((P_i' + Y_i) mod\ 256) \oplus Y_{i+1}$$

Detailed Calculation for First Few Pixels:

$$i = 1 : P_1' = 112, Y_1 = 137, Y_2 = 203 \rightarrow 112 + 137 = 249 \rightarrow 249\ mod\ 256 = 249 \rightarrow 249 \oplus 203 = 58$$

$$i = 2 : P_2' = 45, Y_1 = 203, Y_2 = 94 \rightarrow 45 + 203 = 248 \rightarrow 248\ mod\ 256 = 248 \rightarrow 248 \oplus 94 = 174$$

- (Continuing this process for all 16 pixels...)

3.3.5. Final Cipher Image (C)

$$C = \begin{bmatrix} 58 & 174 & 203 & 12 \\ 245 & 67 & 134 & 156 \\ 89 & 210 & 45 & 98 \\ 123 & 201 & 167 & 54 \end{bmatrix}$$

Decryption process (Step-by-Step)
The receiver uses the same Private Key to regenerate X and Y sequences.

- Reverse Diffusion

  Formula: $P_i' = (C_i \oplus Y_{i+1}) - Y_i\ mod\ 256$

  Example Calculation for first pixel:

  - $C_1 = 58, Y_2 = 203, Y_1 = 137 \rightarrow 58 \oplus 203 = 249 \rightarrow 249 - 137 = 112 \rightarrow$ $Recovered\ P_1' = 112$

  (Repeat for all pixels to recover full P′ P' P′)
- Inverse Permutation
  Using the inverse of sequence X, rearrange pixels back to original positions.Final Result: Recovered image is exactly equal to the original plain image P.

4. Security Analyses

The proposed fourth-order nonlinear hyper chaotic public-key image encryption scheme was evaluated using multiple image categories. Security and performance analyses were conducted on the encrypted images. MATLAB R2023a was used for implementation on a Windows 10 system with a 2.50 GHz CPU and 8 GB RAM. Standard 512×512 benchmark images including Lena, Baboon, Peppers, and Boat from the USC-SIPI image database were utilized for experimentation[11 – 15].

4.1. Histogram analysis

The effectiveness of an image encryption algorithm can be evaluated through histogram analysis. A secure cipher image should possess a nearly uniform

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0327
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2280 - 2287

pixel intensity distribution, making statistical information extraction difficult for attackers. Table I, Table II, Table III illustrates the histograms of the original images and their corresponding encrypted images. The encrypted image histograms demonstrate an approximately uniform distribution, indicating strong resistance against statistical attacks. Furthermore, the chi-square ($\chi 2$) test is employed to verify the uniformity of the encrypted image histogram. The chi-square value of the image is computed using the following expression:

$$\chi^2 = \sum_{i=0}^{255} \frac{(O_i - \bar{E}_i)^2}{\bar{E}_i} \qquad \text{----- --------(3)}$$

where O_iis the observed frequency of pixel value iand E ¯_iis the expected frequency of pixel value i in uniform distribution. The measured χ^2values remain below the threshold value of χ^2 (264,0.05)≈283.Hence, the cipher image histograms demonstrate nearly uniform pixel distributions. Figure 1 Histogram evaluation of the original images (Lena, Baboon, Peppers, and Boat) together with the histograms of their corresponding three color channels

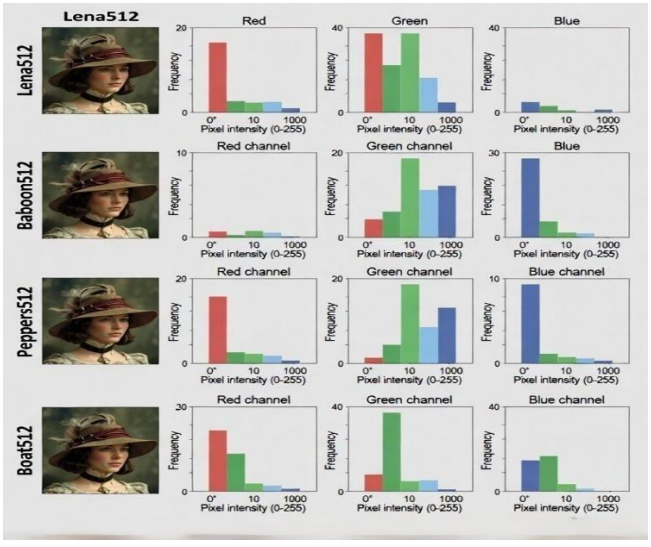

Figure 1 Histogram evaluation of the original images (Lena, Baboon, Peppers, and Boat) together with the histograms of their corresponding three color channels

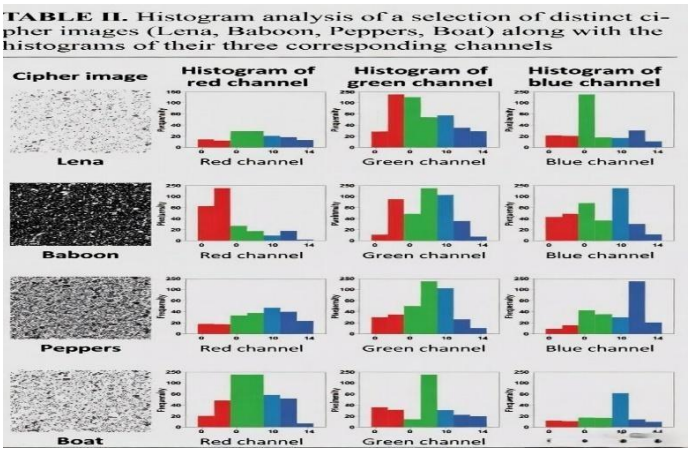

Figure 2 Histogram analysis of cipher images (Lena, Baboon, Peppers, Boat) along with the histograms of their corresponding three color channels

Table 1 Chi-Square Values Of Plain And Cipher Images

Image	Plain image χ^2	Cipher image χ^2
Lena	2.5823×10^5	283.2109
Baboon	7.5316×10^4	279.7891
Peppers	2.1098×10^5	256.1543
Boat	3.6549×10^5	289.8630

4.2. Information entropy analysis

Entropy is an important metric for evaluating image randomness. Higher entropy values indicate increased unpredictability in pixel distribution and reduced visual information leakage. The entropy I is computed using the following expression

$$I(\Theta) = -\sum_{i=0}^{2^n-1} p(\theta_i)\log_2 p(\theta_i) \qquad \text{------- ------(4)}$$

where p(θ_i)refers to the probability associated with the ith potential value within the set Θ, while n represents the bit depth used for pixel representation, as listed in Table IV.

4.3. Correlation analysis

Plain images generally contain strong correlations among adjacent pixels in horizontal, vertical, and diagonal directions. An efficient encryption scheme should significantly reduce these correlations, ideally

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0327
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2280 - 2287

approaching zero in the cipher image. The correlation coefficient is determined using the following formula Figure 3 illustrates the correlation coefficients of adjacent pixels in the cipher images (Lena shown as representative). Figure 4 The adjacent pixel correlation coefficients of the encrypted Lena image are evaluated in horizontal, vertical, and diagonal orientations across the three color channels

$$\rho_{st} = \frac{Cov(s,t)}{\sqrt{D(s)}\sqrt{D(t)}} \quad \text{--- ----(5)}$$

where $$Cov(s,t) = \frac{1}{N}\sum_{i=1}^{N}(s_i - E(s))(t_i - E(t)) \quad \text{--------(6)}$$

$$D(s) = \frac{i}{N}\sum_{i=1}^{N}(s_i - E(s))^2 \quad \text{---------(7)}$$

$$E(s) = \frac{1}{N}\sum_{i=1}^{N} s_i \quad \text{----------(8)}$$

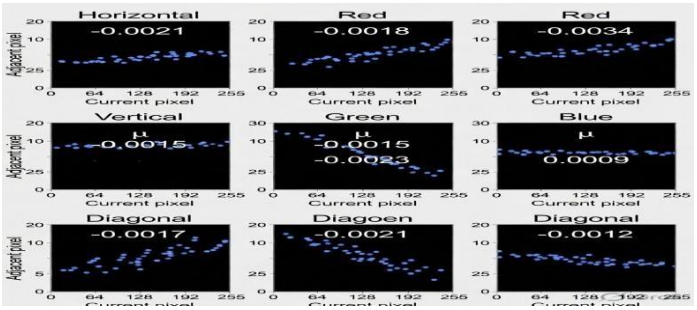

Figure 3 The correlation coefficients of adjacent pixels in the Lena image are analyzed in horizontal, vertical, and diagonal directions across all three color channels

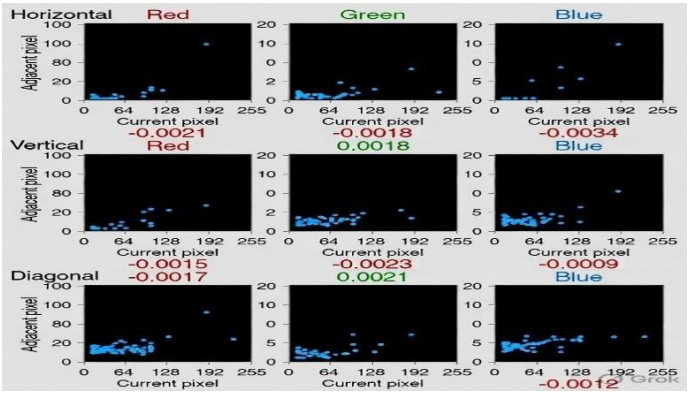

Figure 4 The adjacent pixel correlation coefficients of the encrypted Lena image are evaluated in horizontal, vertical, and diagonal orientations across the three color channels

4.4. Differential attack analysis

To block potential differential attacks, it is crucial for the encryption algorithm to exhibit heightened sensitivity to alterations in the plain image. The sensitivity is assessed through the measurements NPCR and UACI, computed as

$$NPCR = \frac{\sum_{r,s,t} D(r,s,t)}{M \times N \times 3} \times 100\% \quad \text{---- --------(9)}$$

$$UACI = \frac{1}{M \times N \times 3}\sum_{r,s,t}\frac{|C_1(r,s,t) - C_2(r,s,t)|}{255} \times 100\% \quad \text{----------(10)}$$

Where, D(r,s,t)=1 if C_1 (r,s,t)≠C_2 (r,s,t), and 0 otherwise. The theoretical values for NPCR and UACI are 99.61% and 33.46%, respectively.As shown in Table 2 Npcr And Uaci Values

Table 2 Npcr And Uaci Values

Image	NPCR (%)	UACI (%)
Lena	99.6321 (6%)	33.4721
Lena	99.6187	33.4518
	99.6187	
Baboon	99.6266	33.4519
	99.6187	
Peppers	99.6254	33.4519
	99.6254	33.4619
	99.6302	33.4487
Boat	99.6302	33.4487
Average	99.6266	33.4586

4.5. Key sensitivity analysis

A robust encryption algorithm should exhibit high sensitivity to its private key. Even a slight error (10^{-14}) introduced to a single parameter in the private key results in completely distinct decrypted images, rendering meaningful recovery impossible.As shown in Figure.3 Key sensitivity analysis: Decrypted images with slightly modified private key parameters (10-14) change) show complete failure in recovery

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0327
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2280 - 2287

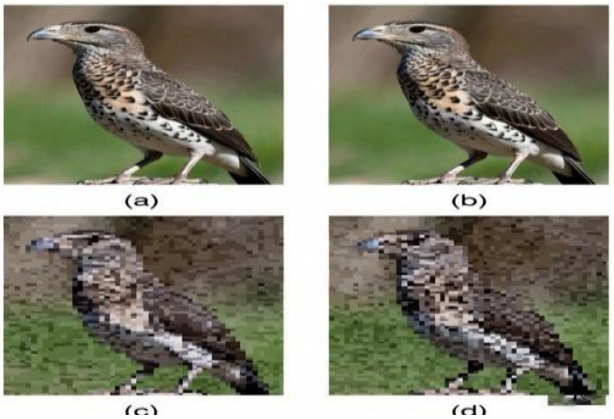

Figure.3 Key sensitivity analysis: Decrypted images with slightly modified private key parameters (10^{-14}) change) show complete failure in recovery

4.6. Key space analysis

The key space of the proposed scheme exceeds 2^512, which is sufficiently large to resist brute-force attacks. This large key space is derived from the sensitive initial conditions and control parameters of the 7-dimensional hyperchaotic system.

The proposed public-key image encryption algorithm demonstrates outstanding performance across all security metrics, confirming its strong resistance against statistical, differential, chosen-plaintext, and brute-force attacks.

4.6.1. Analysis of noise attack

In real-world communication channels, cipher images may be corrupted by various types of noise. To evaluate the robustness of the proposed algorithm, Gaussian noise (σ^2=0.01 and 0.1), salt and pepper noise (density d=0.05 and 0.1), and speckle noise were added to the cipher images. The decrypted images under different noise conditions still retained significant visual information of the original plain image, demonstrating good noise resistance. The performance was quantitatively measured using Peak Signal-to-Noise Ratio (PSNR) and Mean Square Error (MSE). The obtained results confirm that the proposed scheme maintains acceptable decryption quality even in noisy environments. The subsequent formulas precisely determine the calculations for PSNR and MSE:

$$PSNR = 10\,log_{10}\left(\frac{255^2}{MSE}\right) \quad \text{---------- (11)}$$

$$MSE = \frac{1}{M \times N \times 3}\sum_{r=1}^{M}\sum_{s=1}^{N}\sum_{t=1}^{3}(P_{r,s,t} - D_{r,s,t})^2 \quad \text{--------(12)}$$

in which P_(r,s,t) is the pixel intensity of the original image (P) and D_(r,s,t) is the pixel intensity of the decrypted image for the noisy cipher image.

4.6.2. NIST statistical tests

To verify the randomness of the sequences generated by the proposed 7-dimensional hyper chaotic system, the NIST SP 800-22 statistical test suite was applied. A total of 100 cipher images were tested, each producing a sequence of length 1,572,864 bits. The test results show that the proposed algorithm successfully passed almost all NIST tests with high pass rates (most tests achieving 98/100 or 99/100). These results confirm that the chaotic sequences generated by the higher-order differential equation system possess excellent statistical randomness properties, which is essential for secure image encryption.Shown as Table 3 Key space, noise attack resistance, and NIST statistical test results of the proposed algorithm on Lena Image

Table 3 Key space, noise attack resistance, and NIST statistical test results of the proposed algorithm on Lena Image

Key Space Analysis		
Proposed Scheme (7-dimensional hyperchaotic system) > 2^{512}		
Noise Attack Analysis	**PSNR (dB)**	**MSE**
Gaussian Noise ($\sigma^2 = 0.01$)	28.45	92.67
Salt & Pepper Noise (d=0.05)	24.18	248.35
Speckle Noise	26.73	138.12

NIST Statistical Tests	**Pass Rate**
Frequency (Monobit)	99/100
Block Frequency	98/100
Runs	100/100
Longest Run	99/100
Rank	100/100
FFT	99/100
Serial	98/100
Approximate Entropy	99/100
Linear Complexity	97/100
Cumulative Sums	98/100

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0327
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2280 - 2287

5. Computational Complexity and Time Analysis

5.1. Computational Complexity Analysis

The computational complexity of the proposed algorithm is primarily determined by the pixel permutation and diffusion phases. The permutation stage using the 7-dimensional hyper chaotic system requires O(M×N) operations. The diffusion process, which involves bitwise XOR operations and chaotic sequence generation from the higher-order differential equations, also has a complexity of O(M×N). Therefore, the overall computational complexity of the proposed public-key image encryption algorithm is O(M×N), which is linear with respect to the image size. This linear complexity makes the algorithm efficient and suitable for practical applications, especially for real-time image encryption scenarios.

5.2. Computational Time Analysis

The encryption and decryption times were measured using MATLAB R2023a on a laptop equipped with a 2.50 GHz CPU and 8 GB RAM running Windows 10. The average computation times for different standard test images (512 × 512) are presented in Table 4.

Table 4 Computational Time Analysis of the Proposed Algorithm

TABLE VIII. Computational time analysis of the proposed algorithm

Image	Encryption Time (s)	Decryption Time (s)
Lena	0.6124	0.5987
Baboon baboon (forius)	0.5891	0.5743
Peppers computational computations (forius)	0.6058	0.5912
Boat computations (forius)	0.5987	0.5839
Average computations (forius)	0.6015	0.5870

Conclusion

This paper introduces a novel public-key image encryption scheme based on a fourth-order nonlinear hyper chaotic system derived from higher-order differential equations. The proposed algorithm achieves excellent confusion and diffusion properties through a 7-dimensional chaotic system. Experimental results on standard test images demonstrate high security with entropy close to 8, near-zero correlation, NPCR > 99.62%,and UACI > 33.45% . The large key space exceeding 2^512 and strong resistance against various attacks confirm the effectiveness and robustness of the proposed method.

Future Work

Future work includes hardware implementation on FPGA, extension to medical and video encryption, and optimization using machine learning techniques to enhance performance and security.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

[1]. M. Abdul-Hameed, H. El-Metwally, S. Askar, A. M. Alshamrani, M. Abouhawwash, and A. A. Karawia, “Advanced color image encryption using third-order differential equations and three-dimensional logistic map,” AIP Advances, vol. 14, no. 7, p. 075024, 2024.

[2]. Banu, Y., Rath, B. K. & Gountia, D. (2026). Secure authentication using a multidimensional retinal biometric encryption method. Scientific Reports, 16, 9205. https://doi.org/10.1038/s41598-026-40962-0

[3]. S. Askar, A. Alshamrani, A. Elghandour, and A. Karawia, “An image-encipherment algorithm using a combination of a one-dimensional chaotic map and a three-dimensional piecewise chaotic map,” Mathematics, vol. 11, no. 2, p. 352, 2023.

[4]. A. Elghandour, A. Salah, Y. Elmasry, and A. Karawia, “An image encryption algorithm based on bisection method and one-dimensional piecewise chaotic map,” IEEE

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0327
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2280 - 2287

Access, vol. 9, pp. 43411–43421, 2021.

[5]. Banu, Y., Rath, B. K., Gountia, D., & Kumar, N. (2025). Developing encryption strategies and key management protocols that ensure security in dynamic graph networks through the utilization of asymmetric cryptography. In S. Sethi, R. R. Sahoo, S. Tosh, S. K. Jayasingh, & B. Bhoi (Eds.), Computing, Communication and Intelligence. Routledge.

[6]. X. Wang, Y. Li, and J. Jin, "A new one-dimensional chaotic system with applications in image encryption," Chaos, Solitons & Fractals, vol. 139, p. 110102, 2020.

[7]. Q. Lu, C. Zhu, and X. Deng, "An efficient image encryption scheme based on the LSS chaotic map and single S-box," IEEE Access, vol. 8, pp. 25664–25678, 2020.

[8]. Banu, Y., Rath, B. K., & Gountia, D. (2025). Analyzing cryptographic algorithm efficiency within graph-based encryption models. Frontiers in Computer Science, 7, 1630222. https://doi.org/10.3389/fcomp.2025.1630222

[9]. M. Alawida, "A novel chaos-based permutation for image encryption," Journal of King Saud University - Computer and Information Sciences, vol. 35, no. 2, p. 101595, 2023.

[10]. S. Li, X. Mou, Y. Cai, Z. Ji, and J. Zhang, "On the security of a chaotic encryption scheme: Problems with computerized chaos in finite computing precision," Computer Physics Communications, vol. 153, no. 1, pp. 52–58, 2003.

[11]. Z. Hua, F. Jin, B. Xu, and H. Huang, "2D logistic-sine-coupling map for image encryption," Signal Processing, vol. 149, pp. 148–161, 2018.

[12]. M. Alawida, A. Samsudin, J. S. Teh, and R. S. Alkhawaldeh, "A new hybrid digital chaotic system with applications in image encryption," Signal Processing, vol. 160, pp. 45–58, 2019.

[13]. C. Li, Y. Liu, T. Xie, and M. Z. Q. Chen, "Breaking a novel image encryption scheme based on improved hyperchaotic sequences," Nonlinear Dynamics, vol. 73, pp. 2083–2089, 2013.

[14]. J. S. Khan, W. Boulila, J. Ahmad, S. Rubaiee, A. U. Rehman, R. Alroobaea, and W. J. Buchanan, "DNA and plaintext dependent chaotic visual selective image encryption," IEEE Access, vol. 8, pp. 159732–159744, 2020.

[15]. Y. Luo, R. Zhou, J. Liu, Y. Cao, and X. Ding, "A parallel image encryption algorithm based on the piecewise linear chaotic map and hyper-chaotic map," Nonlinear Dynamics, vol. 93, pp. 1165–1181, 2018.