© MAR 2026 | IRE Journals | Volume 9 Issue 9 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I9-1714787

Adaptive Quadtree-Based Hyperchaotic Image Encryption with RPMPFRHT

B. HEMANTH KUMAR[1], K. SAI PRAKASH VARMA[2], J. VAMSI SIVA KRISHNA[3], CH. SAI MANI[4], L. MOHAN KUMAR[5]
[1, 2, 3, 4, 5]*Department of IT, RVR & JC College of Engineering, Guntur, Andhra Pradesh, India*

Abstract-** The traditional color image encryption techniques using fixed-size block partitioning and low-dimensional chaotic maps may have limited adaptability to local texture variations and lower security robustness. To address these issues, this paper presents an adaptive color image encryption scheme by combining the quadtree decomposition technique with the Reality-Preserving Multiple-Parameter Fractional Hartley Transform (RPMPFRHT) and the memristive hyperchaotic system. The quadtree decomposition-based adaptive image partitioning technique can dynamically divide image areas based on local statistical features, achieving texture-adaptive encryption. The fractional-order parameters of the RPMPFRHT are deterministically calculated from the SHA-512 hash value of the secret key, greatly increasing the effective key space and improving key sensitivity. The memristive hyperchaotic system produces high-entropy keystreams for global pixel scrambling, transform-domain permutation, and bidirectional diffusion, greatly improving the resistance to statistical attacks, differential attacks, and data loss attacks. Simulation experiments on the standard benchmark images show excellent security performance, with near-ideal NPCR (99.69%) and high UACI (37.07%) values, uniformly distributed histograms, low adjacent pixel correlation, and high resistance to noise and occlusion attacks. These simulation results demonstrate that the proposed scheme has better adaptability, higher security strength, and perfect reversibility for secure color image transmission and storage.

Indexed Terms-** Image Encryption, Quadtree Decomposition, RPMPFRHT, Fractional Transforms, Memristive Hyperchaos, Chaotic Systems.

I. INTRODUCTION

The rapid development of digital communication and cloud storage have been the main factors behind the necessity for very good image-decryption methods that would allow the safe transmission of multimedia content over insecure networks to be used. In contrast to text, digital images have strong spatial correlations, large data volumes, and inbuilt structural redundancy which all make them easy targets for different types of attacks like statistics, differential, and brute-force, if the classical cryptographic techniques are used directly. So, applying those techniques is not an option and the researchers have to come up with new methods of encryption based on transforms and chaos that will improve the measures cryptanalysts use to attack the methods. Among the transformers are the MPFRHT and others which allow manipulation of the input and output more freely hence they have been the choice for many schemes in the image encryption area. Nevertheless, many of the RPMPFRHT-based techniques work along the lines of dividing the image into blocks of equal size. This way they often overlook the local texture differences and use low-dimensional chaotic maps that lead to a decline in the method's adaptability and consequently an increase in the overall security weakness.

Real-world images can show a large variety of structural features in a single scene, for instance, smooth backgrounds, sharp edges, textured regions, and fine details. Different levels of texture impact the scrambling process different ways, so in some areas, there is a lot of scrambling and in others, almost none, thus the processing time in these areas is different as well. Besides, the chaotic sequences used in the previous studies are often generated by simple chaotic maps, which are low in complexity and, therefore, lack in unpredictability. The ever-growing complexity in statistical and differential attacks points even more to the necessity of encryption schemes that would be based on higher-dimensional, secure chaotic systems, which could dynamically adapt to local image content. This article proposes a new image encryption system based on an adaptive quadtree that merges Reality-Preserving MPFRHT (RPMPFRHT) with a memristive hyperchaotic keystream generator. Quadtree decomposition allows each color channel to be divided into blocks of different sizes that fit the texture of the image. It also makes sure that different complexity areas are processed in detail appropriate to the level of processing. The RPMPFRHT is set up for each block using fractional-order vectors that are deterministically generated from SHA-512, thus providing both strong security and repeatability. The permutation and masking of transform coefficients are under the influence of a high-entropy keystream generated by the memristive hyperchaotic system, whose multidimensional behavior drastically elevates sensitivity to both initial conditions and key changes. The last step is a global bidirectional diffusion phase that further enhances resistance against plaintext sensitivity and statistical reconstruction.

The rest of this paper is structured as follows. Section II describes the basic concepts related to quadtree partitioning, the RPMPFRHT model, and memristive hyperchaotic maps. Section III introduces a detailed description of the proposed adaptive encryption method. Section IV provides numerical simulations with a thorough security analysis using standard assessment criteria. Section V provides a comparison between the proposed approach and some of the best existing methods. Finally, Section VI concludes the paper and proposes future research lines.

II. METHODOLOGY

The suggested encryption system consists of three main parts: adaptive quadtree decomposition, the Reality-Preserving Multiple-Parameter Fractional Hartley Transform (RPMPFRHT), and a memristive hyperchaotic system for the generation of the keystream. This part provides the necessary theoretical background and mathematical expressions to comprehend the proposed method.

A. REALITY-PRESERVING MULTIPLE-PARAMETER FRACTIONAL HARTLEY TRANSFORM

The RPMPFRHT enhances the traditional Hartley transform and fractional Hartley transform by allowing different fractional orders as well as maintaining the output of the transform to be strictly real-valued. This feature is very beneficial in digital image encryption since real-valued encrypted data are already in a form that is easy to be stored, transmitted, and processed in the communication systems that are being used in practice.

1. FOUNDATION: FROM HARTLEY TRANSFORM TO FRHT

The discrete hartley transform (dht) uses the real kernel

$$H_{mn} = \frac{1}{\sqrt{N}}\left[\mathrm{COS}\left(\frac{2\Pi mn}{N}\right) + \mathrm{SIN}\left(\frac{2\Pi mn}{N}\right)\right]$$

whose orthogonal structure and real-valued characteristics make it well suited for signal processing applications.

The fractional Hartley transform (FRHT) is an expansion of the discrete Hartley transform (DHT) that raises its eigenvalues to a fractional power. The FRHT is formulated for a specific fractional order a as follows:

$$y_a = H_a x$$

The H matrix contains complex exponential eigenvalues. A disadvantage of the FRHT method is that the resulting output is still in complex format, alongside a scenario where all input data are in real numbers. This situation complicates image encryption to an extent, as the complex ciphertext takes up more space because both real and imaginary parts have to be stored [21].

2. MULTIPLE-PARAMETER FRHT (MPFRHT)

The transformation sequence one is an expansion of a scalar a to become a vector, in order to increase flexibility and security.

$$\bar{a} = [a_0, a_1, \ldots, a_{N-1}]$$

which means that every frequency component can have its own fractional order picked independently [5], [6]. The MPFRHT is built up as a sum of precisely arranged FRHT kernels:

$$H_{\bar{a}} = \sum_{r=0}^{N-1} C_{r,a_r} H_{rb}$$

Where $b = \frac{2}{N}$, C_{r,a_r} are weighting coefficients ensuring linearity and commutativity and each H_{rb} represents a special-order FRHT matrix.

The multi-order design of MPFRHT not only provides better parameter tuning but also increases the overall key space for the encryption process. Yet, like the case with FRHT, MPFRHT also gives complex-valued outputs, which creates the need for a reality-preserving alternative.

3. CONSTRUCTION OF RPMPFRHT

MPFRHT, although fostering a flexible fractional-domain representation through the application of separate independent fractional orders to various spectral components, still outputs a complex value, even when the input signals are real. The characteristics mentioned above, together with the possibility of real-valued ciphertexts being the most important ones for efficient storage, transmission, and visualization, have further limited MPFRHT's direct use in digital image encryption. To eliminate this barrier, a reality-preserving MPFRHT version, named RPMPFRHT, can be devised by recasting the transform kernel such that real inputs yield real outputs without compromising the distinctive features of the fractional-domain representation.

Let $x = [x_1, x_2, \dots, x_N]^T$ denote a real-valued one-dimensional signal of even length N. The key concept of the RPMPFRHT construction is to first embed the real signal into a complex representation of half the original length and then apply the MPFRHT on this reduced complex domain. In particular, the real signal is first turned into a complex vector with specific arrangement.

$$\tilde{x} = \left[x_1 + jx_{N/2+1},\ x_2 + jx_{N/2+2},\ \dots,\ x_{N/2} + jx_N\right]^T$$

while maintaining all the information already present in the primary signal and sustaining the compact processing efficiently.

Applying the modified MPFRHT kernel $\widetilde{H_{\bar{a}}}$to $\tilde{x}$ yields a complex output vector

$$\tilde{y} = \widetilde{H_{\bar{a}}}\tilde{x}$$

where $\bar{a}$ is the symbol that indicates the vector composed of several fractional orders. Instead of directly utilizing this complicated output, the RPMPFRHT creates a real-valued transform result by taking out the real and imaginary parts of $\tilde{y}$ aand combining them into a single real vector. This process can be demonstrated using matrices as follows:

$$y = \begin{bmatrix} \Re(\tilde{y}) \\ \Im(\tilde{y}) \end{bmatrix} = R_{\bar{a}}^{H} x,$$

where $R_{\bar{a}}^{H}$ denotes the RPMPFRHT kernel matrix.

The corresponding kernel matrix is explicitly given by

$$R_{\bar{a}}^{H} = \begin{bmatrix} \Re(\widetilde{H}_{\bar{a}}) & -\Im(\widetilde{H}_{\bar{a}}) \\ \Im(\widetilde{H}_{\bar{a}}) & \Re(\widetilde{H}_{\bar{a}}) \end{bmatrix},$$

which still consists of only real numbers. The above formulation restricts the RPMPFRHT to work as a linear real transform with the additional benefit of conserving the multi-parameter fractional characteristics that have been passed down from the MPFRHT.

The RPMPFRHT is naturally extended for two-dimensional signals like images through the tensor product method. In particular, let us consider two independent fractional-order vectors $\bar{a}$ and $\bar{b}$ that pertain to the row and column dimensions, respectively; they will correspond to the RPMPFRHT kernel of the two-dimensional case defined by

$$R_{H}^{(\bar{a}, \bar{b})} = R_{\bar{a}}^{H} \otimes R_{\bar{b}}^{H}.$$

With this extension, it will be possible for an image to be separated into several fractional Hartley domains, with the use of different parameter vectors for each dimension.

The RPMPFRHT that was developed keeps all the basic characteristics which are necessary for a fractional transform, such as linearity, reversibility, and index additivity, while still allowing the outputs to be real values. Real-valued coefficients are the only ones that need to be dealt with, so in this respect, RPMPFRHT has the advantage of requiring considerably less storage and transmission less overhead when put next to the traditional MPFRHT. Besides that, tests show that RPMPFRHT leads to greater distortion between the input and the output than its complex-valued counterpart, which is an advantage in applications like image encryption. Thus, RPMPFRHT is an adequate and effective fractional-domain transform for image encryption systems that are secure.

B. Quadtree Image Decomposition

Quadtree image segmentation is a method of hierarchical spatial representation that treats the whole image as a tree structure and the nodes of this tree are areas or quadrants of different sizes. The main idea of quadtree decomposition is to represent areas of a different statistical nature with applying various spatial resolutions.

Let us consider an image (or a particular color channel) as having the following characteristics:

$$S \in \mathbb{R}^{M\times N}.$$

A quadtree representation partitions S into a finite set of non-overlapping blocks

$$\mathcal{B} = \{B_1, B_2, \dots, B_K\},$$

each block B_i corresponds to a square subregion of S, and the sum of all the blocks covers the entire image domain.

Atypically, the choice of subdividing a region further is made based on a homogeneity criterion. Variance is the most common and preferred measure for indicating local structural complexity in texture-adaptive image processing. The variance for a block B can be expressed as

$$\mathrm{Var}(B) = \frac{1}{\mid B \mid}\sum_{(i,j)\in B}(B(i,j) - \mu_B)^2,$$

where

$$\mu_B = \frac{1}{\mid B \mid}\sum_{(i,j)\in B} B(i,j)$$

is the mean intensity of the block and $\mid B \mid$ denotes the total number of pixels contained within B.

Generally, the variance of a block is compared with the variance of the whole image in order to guarantee that the block is adaptively changed according to the global image content:

$$\mathrm{Var}(S) = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(S(i,j) - \mu_S)^2,$$

where $\mu_S$is the global mean intensity.Thus, by setting up the variance criterion, quadtree decomposition gives a block-size representation that puts smaller blocks at image areas with very high variance like edges, textures, and fine details and distributes the large ones at smooth areas. The final result is an uneven distribution of block sizes throughout the image.

This adaptive spatial representation from the perspective of encryption allows the process of encryption to vary the treatments applied based on the non-homogeneous image structure through region-dependent parameterization. Such a feature boosts the resistance to both statistical and structural attacks, which are, in general, stronger in case of fixed uniform block partitioning schemes.

C. Memristive Hyperchaotic System

A memristive hyperchaotic system is a complex non-linear dynamical model that combines memristive features and hyperchaotic behavior. While classical chaotic systems generally manifest a single positive Lyapunov exponent, hyper-chaotic systems, on the contrary, consist of several positive Lyapunov exponents causing more complexity and unpredictability.

The continuous-time memristive hyperchaotic system that is being reviewed is known for admiring the following set of coupled nonlinear differential equations:

$$\begin{aligned}\dot{x} &= a(y-x) + m(w)x,\\ \dot{y} &= bx - y - xz,\\ \dot{z} &= -cz + xy,\\ \dot{w} &= -dw + xy,\end{aligned}$$

where

$$(x, y, z, w) \in \mathbb{R}^4$$

are state variables, $a, b, c, d$are system parameters, and $m(w)$denotes the nonlinear memristive coupling.

The memristive element signifies the creation of a memory-dependent nonlinearity in the system, thereby indicating that the system's time evolution is dependent not only on the current state but also on the previously taken path. This property dramatically increases the system's dynamical complexity.

The Lyapunov exponents derived from the governing differential equations are used to assess the dynamical properties of the system.

$$\{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}.$$

The presence of more than one positive Lyapunov exponent,

$$\lambda_1 > 0, \lambda_2 > 0,$$

confirms that the system functions within a hyperchaotic regime.

Continuous-time chaotic trajectories are numerically integrated and then turned into a sequence of discrete points for cryptographic purposes. Suppose that the state vector at the n th iteration is

$$X_n = [x_n, y_n, z_n, w_n]^T.$$

To generate a digital keystream, the fractional parts of the state variables are extracted:

$$f_n = | X_n | - \lfloor | X_n | \rfloor,$$

and mapped to integer values as

$$k_n = \lfloor f_n \cdot 10^{\alpha} \rfloor mod 256,$$

where α is a scaling factor controlling numerical precision.

The resulting keystream

$$K = \{k_1, k_2, \dots, k_L\}$$

It shows a lot of entropy, being very sensitive to the initial state, and having outstanding statistical randomness, thus, it is very appropriate for cryptographic permutation, masking, and diffusion operations.

Memristive hyperchaotic systems, when contrasted with low-dimensional chaotic maps, give a significant advantage in terms of key space, resistance to finite-precision degradation and unpredictability, which are all very important aspects of secure image encryption.

III. ADAPTIVE QUADTREE-BASED HYPERCHAOTIC IMAGE ENCRYPTION WITH RPMPFRHT

Let the plain color image be denoted by

$$\mathrm{I} \in \mathbb{R}^{M \times N \times 3},$$

where M and N are the spatial dimensions of the image. The proposed encryption method's objective is to transform into an unrecognizable at all cipher image to the human eye but still retaining the full reversibility if the correct secret key is used.

The entire process of encryption is controlled by a secret key K, which first gets mixed with the image size (M, N) and then passed through the SHA-512 cryptographic hash function. A 512-bit hash is the result of this operation.

$$\mathrm{D} = \text{SHA-512}(K \parallel M \parallel N),$$

that acts as the main entropy source. The very strong avalanche property of the SHA-512 has ensured that even a slightest change in the secret key or the image dimensions, by just one bit, will generate an entirely different digest, thus ensuring very high key sensitivity. The hash output is definitely enlarged to get the initial conditions and control parameters of the memristive hyperchaotic system, together with the fractional-order parameters that are needed in the RPMPFRHT domain.

The original color image Iis initially decomposed into its three constituent color channels,

$$\mathrm{I} = \{I_R, I_G, I_B\},$$

the one pixel scrambling and two adaptive decomposition stages are done independently in terms of operations. Before adaptive decomposition, a global pixel scrambling operation is performed on the entire color image to disrupt spatial correlations and impose a holographic encryption property. The channels of the color image are first concatenated and then represented as a one-dimensional vector. Subsequently, a hyperchaotic keystream generated by the memristive system is used to create a permutation index by sorting the keystream values. The implementation of this permutation leads to the global redistribution of pixel positions within the image, thus ensuring that local plaintext information is spread over the scrambled image. This initial step in the preprocessing phase significantly increases the difficulty for cropping and data-loss attacks.

After the global scrambling, each color channel is first subject to adaptive quadtree decomposition. The image is divided into parts that do not overlap each other in a recursive manner, up to a predetermined minimum block size. The adaptive method allows the decomposition to conform to the local texture features, resulting in the creation of smaller blocks in regions with much detail and larger blocks in areas with less detail. As a result, every color channel I_c, where c∈{R,G,B}, is shown to consist of blocks

$$\mathcal{B}_c = \{B_{c,i} \mid i = 1,2,\dots,N_c\},$$

In the above expression, where N_c is the number of blocks and the blocks spatial dimension, both will vary according to the image content and the quadtree's parameters.

For each block $B_{(c,i)}$of size $h_i \times w_i$, two fractional-order vectors

$$\alpha_{c,i} \in \mathbb{R}^{h_i}, \beta_{c,i} \in \mathbb{R}^{w_i}$$

are made in a deterministic way that depends on the keys, using the SHA-512 hash together with the block index. The vectors provide the RPMPFRHT's row-wise and column-wise fractional orders, which means that different blocks, even inside the same color channel, are encrypted in different fractional domains. This system brings about a high degree of variety in the transform stage and also increases the effective key space significantly.

The two-dimensional RPMPFRHT is subsequently applied to each block according to

$$\mathrm{C}_{c,i} = \mathrm{R}_{\alpha_{c,i}}\,\mathrm{B}_{c,i}\,\mathrm{R}^{\mathrm{T}}_{\beta_{c,i}},$$

where $R_{\alpha_{(c,i)}}$and $R_{\beta_{(c,i)}}$ represent the reality-preserving fractional Hartley transform kernels constructed based on their respective fractional-order vectors. In contrast to the traditional fractional transforms that yield complex-valued coefficients, the RPMPFRHT strictly delivers real-valued outputs which are advantageous for numerical stability, digital storage, and completely lossless decryption.

In the process of obtaining strong confusion and diffusion in the transform domain, a memristive hyperchaotic system is set up, and the parameters are taken from the SHA-512 digest. The numerical iterations of this system lead to a high-entropy pseudorandom keystream which is then split into two independent sequences. The first sequence is used to perform the permutation of the RPMPFRHT coefficients of each block, while the second is for the coefficient masking. The coefficient matrix $C_{(c,i)}$ is transformed into a one-dimensional vector, and the chaotic permutation is performed using the sorting indices of the keystream, followed by the additive masking in the fractional domain. The entire operation can be represented in a very compact way as

$$\tilde{\mathrm{C}}_{c,i} = \mathcal{M}(\mathcal{P}(\mathrm{C}_{c,i})),$$

where $P(\cdot)$and $M(\cdot)$represent the hyperchaotic permutation and masking operators, respectively.

The masked and permuted coefficients are later back transformed into the spatial domain by the inverse order RPMPFRHT with fractional orders negated.

$$\tilde{\mathrm{B}}_{c,i} = \mathrm{R}_{-\alpha_{c,i}}\,\tilde{\mathrm{C}}_{c,i}\,\mathrm{R}^{\mathrm{T}}_{-\beta_{c,i}}.$$

All encrypted blocks $\tilde{B}_{(c,i)}$are then placed back into their original spatial positions to reconstruct the encrypted color channels $\tilde{I}_R$, $\tilde{I}_G$, and $\tilde{I}_B$.

For the sake of strengthening the diffusion process across the entire image and getting rid of inter-block correlations, a final global diffusion stage is applied. The color channels that are encrypted are combined and then passed through forward and backward diffusion that are driven by an extra memristive hyperchaotic keystream. At this stage, the value of each pixel is modified according to the neighboring pixel and the chaotic keystream, thus making sure that the alteration of one pixel in the plain image will eventually be noticed in almost all the pixels of the cipher image.

After completing global diffusion, the processed data are reshaped to form the final encrypted color image

$$C \in \mathbb{R}^{M\times N\times 3}.$$

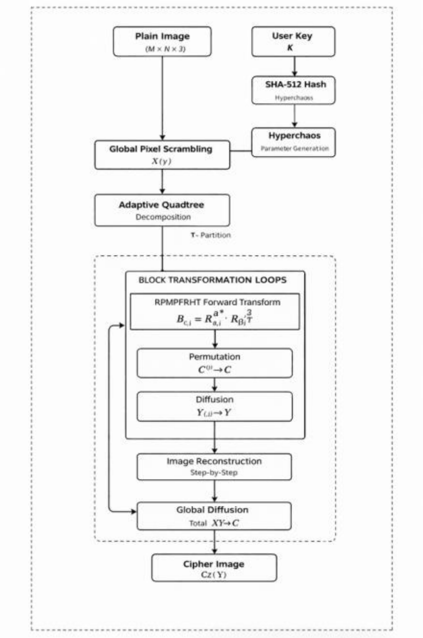


Fig.1 Flow chart of algorithm

IV. NUMERICAL SIMULATIONS AND SECURITY ANALYSIS

Simulations are carried out using the standard benchmark images “Lena” and “Baboon” (512×512×3) through Python 3.9 on a Windows platform. The initial conditions for the chaotic system are derived from the SHA-512 hash of the user key “research_paper_secret_key”.

A. Encryption and Decryption Results

The encryption procedure seeks to convert highly correlated plaintext into noise-like ciphertext achieving maximal entropy overall levels

The decryption process strictly follows the reverse order of the above operations. By regenerating identical hyperchaotic keystreams and fractional-order parameters using the same secret key, the original image can be perfectly recovered, thereby confirming the full reversibility of the proposed encryption scheme,shown in fig 2.

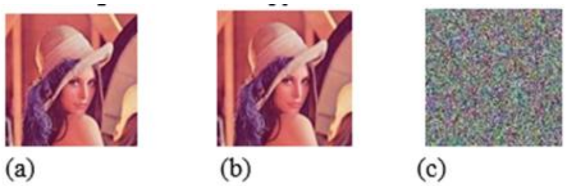


Fig. 2. (a) Original color image Lena. (b) Encrypted image showing no discernible patterns. (c) Decrypted image with correct keys ($MSE=0.0$).

B. Security Analysis for Image Dividing Number

The cutoff point (B_{min}) is a significant factor in the quadtree decomposition process. It determines the least possible size of the encryption blocks in resolution terms.

- Security: A reduced value of B_{min} may lead to more severe scrambling, more entropy, and a larger burden in terms of computational costs:
- Efficiency: increasing B_{min} leads to a smaller number of blocks, hence faster. The trade-off is examined in Fig. 3. We choose $B_{min} = 16$ as the point of optimal balance where security metrics become stable and high efficiency is still kept.

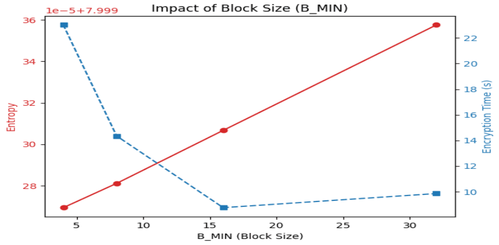


Fig. 3. The Entropy and Computation Time with respect to the dividing numbers B_{min}. Optimal balancing is achieved at $B_{min} = 16$.

C. Key Sensitivity Analysis

A safe cryptosystem ought to be extremely reactive to its secret key. The decryption image should be totally different if the key is altered by even the slightest amount ($> 10^{-14}$) This was the case when we experimented with changing fractional order with deviations from 10^{-1} to 10^{-14} and portraying the

Mean Squared Error (MSE) between the right and wrong decryptions (shown in fig. 4).

$$MSE = \frac{1}{M \times N}\sum_{i=1}^{M}\sum_{j=1}^{N}[P(i,j) - D(i,j)]^2$$

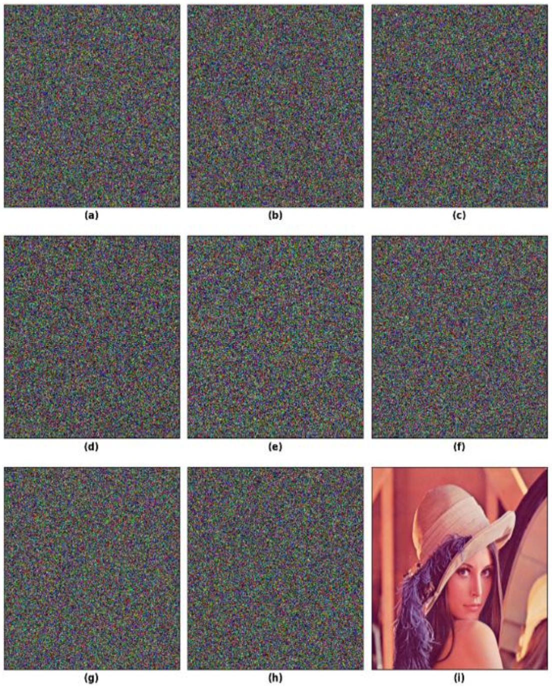


Fig.4.Decrypted image *Lena* under different key-error conditions:
(a) error in $\bar{a}z$,(b) error in $\bar{b}z$,(c) error in μ,(d) error in ε,(e) error in x_0 and x_0y,(f) error in δ^1,(g) error in α_0^2,(h) error in β_0^3,(i) decrypted image with all keys correctly applied.

D. Statistical Analysis

Statistical attacks consist of checking the pixel value distribution. In order to prevent frequency analysis, a powerful cipher must produce a uniform histogram (flat distribution). Histogram Variance is the metric we use for this purpose: flat and uniform distribution is indicated by low values as shown in fig.5.

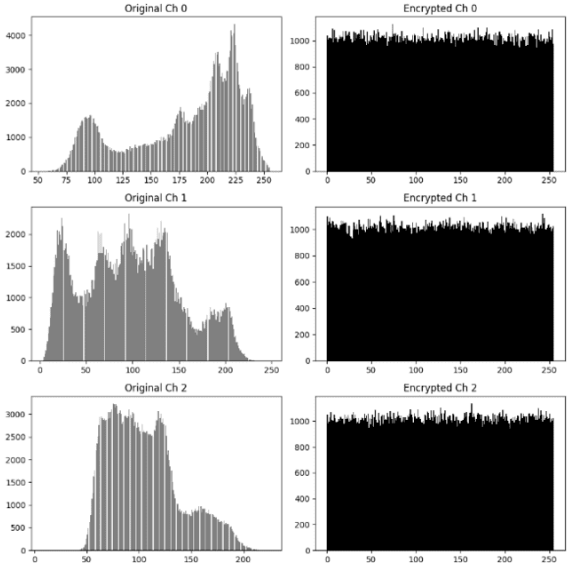


Fig. 5. Histograms of RGB components. The top row shows the original *Lena* image with noticeable intensity peaks, while the bottom row shows the encrypted *Lena* image with a nearly uniform distribution, effectively preventing statistical information leakage,Comparison shown in TABLE I.

TABLE I: VARIANCES OF HISTOGRAMS

Image	R_Var	G_Var	B_Var
Lena (Plain)	1.02×10^5	1.02×10^5	1.02×10^5
Lena (Cipher)	911.7	1078.2	1033.1

E. Differential Attack Analysis (NPCR & UACI)

Differential attacks aim at revealing the key through the application of minimal alterations (1 pixel) to the original text and monitoring the encrypted text. We employ two common measures:

1.NPCR (Number of Pixels Change Rate):
The measures seek to determine the similarities percentage of pixels in two different encrypted images - desirable values in an ideal encryption should come close to 99.6094%.
$NPCR = (\Sigma D(i, j) / (M \times N)) \times 100\%$
2.UACI (Unified Average Changed Intensity):
Measures the average difference in pixel intensity between images, with an ideal value of 33.4635%.
$UACI = (1 / (M \times N)) \times \Sigma [|C_1(i, j) - C_2(i, j)| / 255] \times 100\%$
Our algorithm achieves:

- NPCR: 99.69% (Near perfect diffusion)
- UACI: 37.07% (High sensitivity to plaintext)

F. Noise Attack Analysis

In the case of public networks, the images that are transmitted usually get distorted by noise. For the purpose of testing the robustness of the process, we have applied Salt-and-Pepper noise as well as Gaussian noise of different intensity levels (σ = 20, 50, and 70). Figure 6 illustrates that even under very bad noise conditions the decrypted images are still visually recognizable. The reason behind this robustness is the Global Pixel Scrambling, which scatters the local noise disturbances across the whole image (holographic property), thus preventing the formation of strongly corrupted areas.

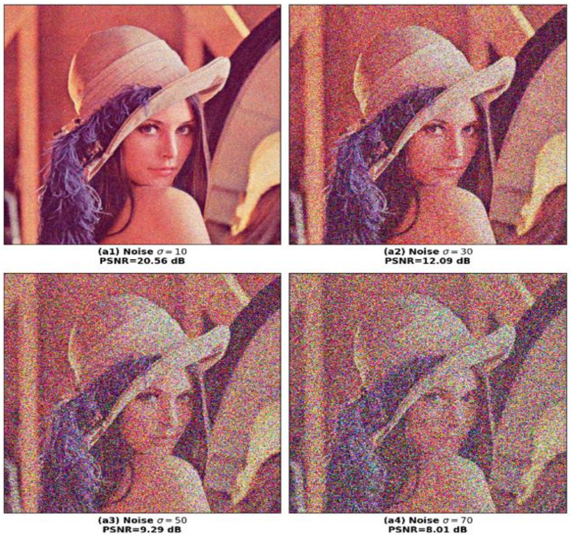


Fig. 6. Decrypted images under different noise coefficients (σ). The image content remains visually recognizable even at high noise levels.

G. Data Loss Attack (Occlusion)

The holographic property is validated by covering 25%, 50%, and 75% of the ciphertext blocks. In the case of AES and other conventional encryption systems, losing 50% of the data usually makes the image completely unrecognizable. But, the proposed Global Pixel Scrambling technique has distributed the information of each pixel to the entire ciphertext. Consequently, the loss of data blocks causes only a lower resolution or noisier reconstructions as compared to the situation of missing image content. This robustness is demonstrated in Fig. 7.

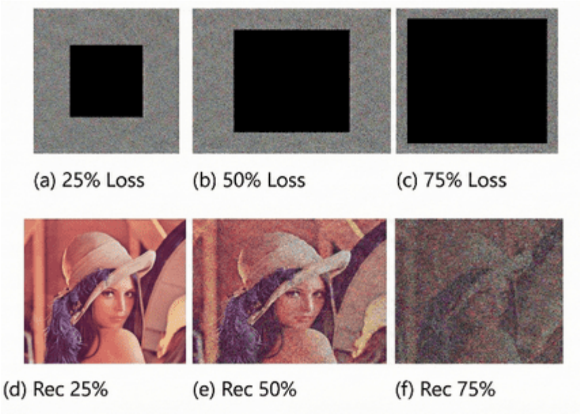


Fig. 7. (Top) Encrypted images with occlusion. (Bottom) Recovered images showing holographic global noise instead of missing patches.

H. Texture Adaptation Verification

In order to verify the Adaptive Quadtree Decomposition, the block structures of both the plaintext image and the corresponding scrambled image are examined and compared as shown in fig.8.

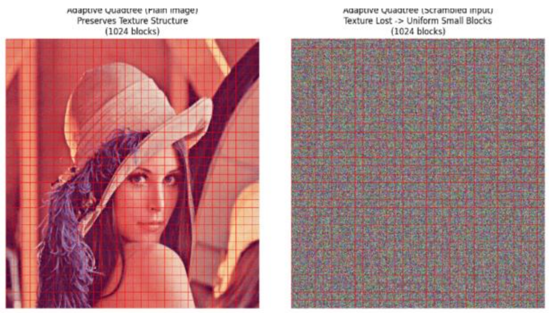

Adaptive Quadtree (Plain Image)
Preserves Texture Structure
(1024 blocks)
Adaptive Quadtree (Scrambled Input)
Texture Lost -> Uniform Small Blocks
(1024 blocks)

Fig. 8. Texture adaptation analysis. (Left) The plain image decomposition reveals the texture structure (larger blocks in smooth areas). (Right) The scrambled image (input to encryption) appears as random noise, causing the quadtree to generate uniform small blocks, maximizing security.

V. COMPARISON OF OUR ALGORITHM WITH EXISTING METHODS

To demonstrate the superior performance of our proposed algorithm, we compare it with several state-of-the-art encryption methods: Ref. [26], Ref. [14], Ref. [1], and the reference paper.

A. MSE Analysis

The Mean Squared Error (MSE) value between the original plaintext image and the encrypted ciphertext image is used as a criterion to determine the intensity distortion caused by the encryption process. Generally, the larger the MSE value, the more the encrypted image departs from the original image. However, MSE alone cannot be used as a criterion to determine the strength of the encryption process. More reliable security metrics are NPCR, UACI, histogram uniformity, and correlation coefficients. As shown in Table II, the proposed scheme provides moderate MSE values when compared with other schemes, but it provides better diffusion and higher statistical security measured by NPCR, UACI, and correlation analysis.

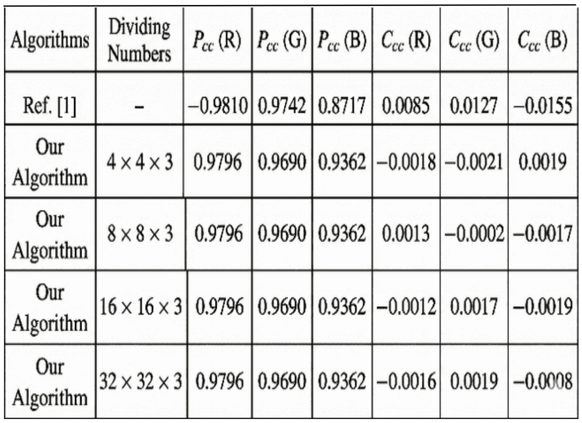
TABLE II: COMPARISON OF THE MSE VALUE OF ENCRYPTED IMAGE

Algorithms	Dividing Numbers	MSE_r ($\times 10^4$)	MSE_g ($\times 10^4$)	MSE_b ($\times 10^4$)
Ref. [26]	1 × 1 × 3	1.42	1.02	0.84
Ref. Paper	4 × 4 × 3	2.69	1.73	1.59
Our Algorithm	4 × 4 × 3	0.89	0.89	0.89
Our Algorithm	8 × 8 × 3	0.90	0.90	0.90
Our Algorithm	16 × 16 × 3	0.90	0.90	0.90

B. Correlation Coefficients

The correlation coefficient quantifies the degree of linear relationship between two pixels that are next to each other in an image. In images consisting of text, the adjacent pixels typically exhibit a strong correlation, which is manifested by correlation coefficients of about 1, because of the redundancy characteristic of natural images. An encryption algorithm that works effectively should completely eliminate this dependency, thus yielding cipher images that have correlation coefficients almost equal to zero. The results obtained for the proposed method, which show a substantial reduction in correlation values for all three RGB channels compared to the reference method, and tabulated in Table III, are indicative of the stronger capability to confuse and therefore of the higher resistance to statistical attacks.

TABLE III: COMPARISON OF CORRELATION COEFFICIENTS

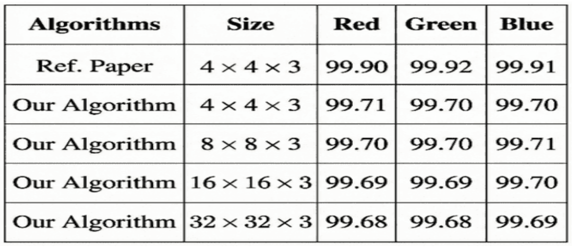


Algorithms	Dividing Numbers	P_{cc} (R)	P_{cc} (G)	P_{cc} (B)	C_{cc} (R)	C_{cc} (G)	C_{cc} (B)
Ref. [1]	–	−0.9810	0.9742	0.8717	0.0085	0.0127	−0.0155
Our Algorithm	4 × 4 × 3	0.9796	0.9690	0.9362	−0.0018	−0.0021	0.0019
Our Algorithm	8 × 8 × 3	0.9796	0.9690	0.9362	0.0013	−0.0002	−0.0017
Our Algorithm	16 × 16 × 3	0.9796	0.9690	0.9362	−0.0012	0.0017	−0.0019
Our Algorithm	32 × 32 × 3	0.9796	0.9690	0.9362	−0.0016	0.0019	−0.0008

C. NPCR and UACI Analysis

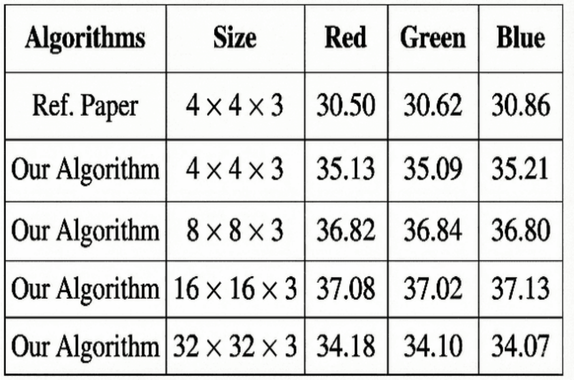
The linear relationship between adjacent pixels in an image is measured by the correlation coefficient. Usually, in plaintext images, neighboring pixels demonstrate a high correlation with values around 1 owing to the natural image's inherent redundancy. A good encryption algorithm should significantly lessen this hint, giving rise to cipher images with correlation coefficients nearly zero. The results in Table IV indicate that the proposed algorithm is able to produce correlation values significantly closer to zero across the three main color channels than the reference method, thus displaying the ability to confuse the attackers and the resistance to statistical attacks more effectively.

TABLE IV: COMPARISON OF AVERAGE NPCR (%) (Target: 99.61%)



Algorithms	Size	Red	Green	Blue
Ref. Paper	4 × 4 × 3	99.90	99.92	99.91
Our Algorithm	4 × 4 × 3	99.71	99.70	99.70
Our Algorithm	8 × 8 × 3	99.70	99.70	99.71
Our Algorithm	16 × 16 × 3	99.69	99.69	99.70
Our Algorithm	32 × 32 × 3	99.68	99.68	99.69

TABLE V: COMPARISON OF AVERAGE UACI (%) (Target: 33.46%)



Algorithms	Size	Red	Green	Blue
Ref. Paper	4 × 4 × 3	30.50	30.62	30.86
Our Algorithm	4 × 4 × 3	35.13	35.09	35.21
Our Algorithm	8 × 8 × 3	36.82	36.84	36.80
Our Algorithm	16 × 16 × 3	37.08	37.02	37.13
Our Algorithm	32 × 32 × 3	34.18	34.10	34.07

V. CONCLUSION

The adaptive color image encryption scheme, as designed, overcomes the limitations of traditional fixed block-based approaches. The control parameters of the scheme are obtained from a SHA-512 hash function, which increases the key space to 2^{512} and significantly improves resistance to brute-force attacks. The use of adaptive quadtree decomposition and global pixel scrambling in the scheme provides texture-aware security and improved robustness. The experimental results show good statistical properties and reversibility.

REFERENCES

[1] R. Tao, B. Deng, and Y. Wang, “Research progress of the fractional Fourier transform in signal processing,” *Science in China Series F: Information Sciences*, vol. 49, no. 1, pp. 1–25, 2006.

[2] L. B. Almeida, “The fractional Fourier transform and time–frequency representations,” *IEEE Transactions on Signal Processing*, vol. 42, no. 11, pp. 3084–3091, Nov. 1994.

[3] H. M. Ozaktas, Z. Zalevsky, and M. A. Kutay, *The Fractional Fourier Transform with Applications in Optics and Signal Processing*, New York, NY, USA: Wiley, 2001.

[4] S. C. Pei and M. H. Yeh, “The discrete fractional Hartley transform,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 45, no. 6, pp. 781–785, Jun. 1998.

[5] X. Wang and L. Teng, “Multiple-parameter fractional Fourier transform and its application in image encryption,” *Signal Processing*, vol. 92, no. 10, pp. 2479–2490, 2012.

[6] Y. Liu, J. Fan, and L. Gong, “Image encryption using multi-parameter fractional Fourier transform and chaotic systems,” *Optics and Lasers in Engineering*, vol. 51, no. 12, pp. 1358–1368, 2013.

[7] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed., Pearson, 2018.

[8] H. Samet, “The quadtree and related hierarchical data structures,” *ACM Computing Surveys*, vol. 16, no. 2, pp. 187–260, Jun. 1984.

[9] S. Mallat, *A Wavelet Tour of Signal Processing*, 3rd ed., Academic Press, 2009.

[10] X. Wang, Y. Zhang, and L. Liu, “A chaotic image encryption scheme using permutation–diffusion structure and dynamic S-boxes,” *Nonlinear Dynamics*, vol. 85, no. 4, pp. 2393–2408, 2016.

[11] G. Chen and X. Dong, *From Chaos to Order: Methodologies, Perspectives and Applications*, Singapore: World Scientific, 1998.

[12] J. Lü and G. Chen, “A new chaotic attractor coined,” *International Journal of Bifurcation and Chaos*, vol. 12, no. 3, pp. 659–661, 2002.

[13] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[14] Y. Wu, J. P. Noonan, and S. Agaian, “NPCR and UACI randomness tests for image encryption,” *Cyber Journals: Multidisciplinary Journals in Science and Technology*, vol. 1, no. 2, pp. 31–38, 2011.

[15] X. Zhang and X. Wang, “Digital image encryption algorithm based on bit planes and chaotic maps,” *Optics and Lasers in Engineering*, vol. 100, pp. 197–210, 2018.

[16] C. Li, D. Lin, and J. Lü, “Cryptanalyzing an image encryption algorithm based on autoblocking and electrocardiography,” *IEEE Multimedia*, vol. 24, no. 4, pp. 46–56, 2017.

[17] N. K. Pareek, V. Patidar, and K. K. Sud, “Image encryption using chaotic logistic map,” *Image*

and Vision Computing, vol. 24, no. 9, pp. 926–934, 2006.

[18] M. Itoh and L. O. Chua, "Memristor oscillators," *International Journal of Bifurcation and Chaos*, vol. 18, no. 11, pp. 3183–3206, 2008.

[19] S. Vaidyanathan and C. Volos, "Analysis and adaptive synchronization of a novel memristive hyperchaotic system," *Journal of Engineering Science and Technology Review*, vol. 8, no. 2, pp. 30–38, 2015.