scientific reports

OPEN

A blockchain-integrated chaotic fractal encryption scheme for secure medical imaging in industrial IoT settings

Saba Inam[1]✉, Shamsa Kanwal[1], Mamoona Batool[1], Shaha Al-Otaibi[2] & Mona M. Jamjoom[3]

The increasing adoption of smart cameras and image sensors in industrial and medical applications necessitates robust visual data security solutions. The industrial Internet of Things (IoT) introduces unique security challenges, particularly due to third-party involvement, which undermines traditional security mechanisms. This study presents a three-layered encryption scheme integrating novel blockchain technology with chaotic fractal image encryption scheme to address these challenges. The encryption process combines an S-box generated from the May map for pixel substitution with fractal-based key generation using a logistic map-driven Sierpinski triangle and incorporates a Chebyshev map-based diffusion step for enhanced randomness and security. Extensive testing, including key sensitivity analysis, entropy calculations (average entropy: 7.9998), NPCR (99.92%), UACI (33.31%), and PSNR values (29.74 dB for encrypted images), validates the scheme's robustness. The results confirm high resistance to differential and brute-force attacks, making the scheme highly suitable for securing sensitive medical images in IoT environments while ensuring confidentiality and integrity.

Keywords Chaotic map, Image encryption, Fractal-based key generation, Blockchain technology, May map, S-box, Medical image security, IoT security

The rapid expansion of WiFi and 5G has led to the frequent transfer of large digital images across various communication networks. However, the increased frequency of these transfers has made the images more vulnerable to tampering and theft. To ensure the security and privacy of these images before transmission, methods like image encryption and digital watermarking are commonly used. Image encryption is the most straight forward method for protecting digital images[1]. During the permutation stage, pixel positions are randomly rearranged, while the diffusion stage alters the pixel values of the original image. Techniques used in image encryption include frequency domain transformations, chaos, evolutionary algorithms, neural networks (NNs), and DNA coding. Chaos is particularly popular due to its sensitivity and inherent unpredictability, making it a powerful tool in encryption methods[2].

The Internet of Things (IoT) and blockchain technology has offered an advanced solutions for enhancing image encryption techniques. IoT devices can collect and transmit images in real-time making it crucial to secure these data transfers. By embedding encryption protocols directly with IoT devices, images can be protected from unauthorized access as they are captured and transmitted. Blockchain technology can further enhance image encryption by providing a decentralized and immutable ledger for tracking image data. Each transaction involving an image whether it is encrypted or is in its original form, it can be recorded on the blockchain. This ensures that the tampering proof of all the activities related to the stored image are saved on the blockchain, thereby enhancing its security. Combining the use of IoT with blockchain offers a robust framework for securing digital images from capture to transmission[3].

[1]Department of Mathematical Sciences, Fatima Jinnah Women University, The Mall, Rawalpindi, Pakistan. [2]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, 11671 Riyadh, Saudi Arabia. [3]Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, 11671 Riyadh, Saudi Arabia. ✉email: saba.inam@fjwu.edu.pk

IoT devices are interconnected through wireless or wired networks and they provide seamless communication and data exchange. The network connectivity helps the devices to work together and perform multiple tasks. The integration of data analytic, AI, and Machine Learning (ML) algorithms allow the IoT devices to make informed decision and process a large amount of data[4]. The use of intelligence helps enhance the functionality and efficiency of IoT system. IoT ecosystem contains a number of devices with varying hardware platforms, operating system and protocols. This heterogeneity requires robust frameworks to ensure seamless interaction among different devices. Nowadays IoTs are used in smart homes, health care, industrial IoT, smart cities, agriculture, transportation and logistics[5].

Blockchain technology is a revolutionary innovation which has transformed the way data can be recorded verified and shared. It is a decentralized ledger system which ensures transparency security and protection of data on online forums. This decentralized nature of blockchain technology reduces the risk of single point failures, enhancing the systems resilience against attacks. In a blockchain system the transactions are visible to all participants in that network which promotes trust and accountability. Furthermore once a transition is recorded on the blockchain it cannot be altered or deleted which provides us the integrity of data. The key components of blockchain includes blocks, ledger, consensus, cryptographic security and smart contracts. Blockchain contains a series of blocks which are connected with each other and every block contains a list of transaction with time stamp and cryptographic hash of the previous block. This interconnected structure ensures the integrity and order of the data. Blockchain operate on the distributed networks where all the participants also known as nodes maintain a copy of the entire ledger[6]. The distributed ledger eliminates the need for a central authority and reduces the risk related to the single point of failure. The consensus mechanism of a blockchain technology like proof of work (PoW) or proof of stake (PoS) validates the transactions and eliminates the need for central authority making it more easier to use. Blockchain uses cryptographic algorithms and techniques to secure the transaction and protect the integrity of data. Different pairs of public and private keys are used for encryption and creating. Digital signatures are written into courts which automatically enforce and execute when the defined terms are completed[7].

Blockchain technology is now used in cryptocurrencies like Bitcoin. It provides the supply chain management by enhancing the transparency and a temper-proof record of the movement of goods from origin till destination. Blockchain technology is also used in medical in efficiently securing the medical records and improving the patient's care. Other applications of blockchain include financial services, voting system, intellectual property and in real estate[8].

The Blockchain-Orchestrated multi-cloud strategy refers to a decentralized system that uses the blockchain technology to coordinate and optimize the management of resources across multiple cloud server providers such as AWS cloud, azure and Google cloud. We have employed this strategy in our proposed image encryption technique as it integrates the inherent advantages of blockchain with the elasticity of a multi-cloud environment. This strategy facilitates secure resource allocation, efficient data distribution and an enhanced redundancy across different cloud platforms. The multi cloud option mitigates the risk of vendor lock in hence improving the system's resilience and in data integrity through continuous verification process which is governed by the blockchain technology[9]. Besides this, the blockchain orchestrator manages the cryptographic keys and ensures that any transaction or change related to data storage, data retrieval, and data processing are securely logged in and verified. The orchestration layer dynamically optimizes the cloud resource usage and maintains a robust security standard which makes it highly suitable for applications that demand high levels of trust and security in distributed environments.

A research focused on secure and efficient image encryption was proposed which addressed the challenges posed by proliferation of digital images over open networks. The study introduced a 5-stage image encryption algorithm which was designed using Shannons confusion and diffusion principles[10].

In the realm of secure steganography, Alexan et al. Introduced "Stegocrypt", a tri-stage algorithm that enhances data concealment within digital images. This method involved encrypting a secret message using tan logistic map, then converting it to quick response (QR) code which are then decoded back into bit-streams. This scheme used DNA coding in each bit-stream by help of a uniquely-seeded Mersenne Twister key which significantly increased its security[11].

Another novel image encryption technique was proposed which integrated the unique image transformation techniques with the principles of chaotic and hyper-chaotic systems. It combined the unpredictable behaviour of Chua system and the hyper-chaotic nature of the Chen system resulting in an expansive key space of 25,208[12].

Related works

Alexan W. et al., developed an image encryption scheme that extended the hyperchaotic 4D chen system into fractional-order domain to perform encryption on 3 stages involving discrete Fourier transform (DFT), S-box derived from DFT and lastly a Mersenne Twister encryption key. The use of three stages provided with a promising solution for secure image encryption[13].

A novel 3-stage cryptosystem was proposed which utilized a three-stage process which combined chaotic maps and DNA encoding to enhance security. The first stage used a tan variation of logistic map for DNA encoding, the second stage constructed a robust S-box using the Lorenz differential equations and a linear descent algorithm and the third stage applied the original logistic map for further diffusion. This approach offered high security and computational efficiency[14].

A study utilized a fractional-order 4D hyperchaotic Chen map combined with sine chaotic map and a hybrid DNA coding algorithm. This approach demonstrated robustness against various attacks[15]. Additionally, an image encryption algorithm designed for color medical images employed Fibonacci Q-matrices, an S-box in Galois field 2^8 and a hyperchaotic system modeling a memristive coupled neural network[16].

In recent years the integration of smart cameras and image sensors in industrial processes has been increasingly prevalent, especially for quality assurance. However the widespread in use of these technologies has also raised concerns regarding the security of the data, especially in the context of the industrial internet of things (IIoT). Traditional cryptographic solutions often struggle to provide adequate protection in these environments due to the involvement of third-party excess which is a threat to security. Blockchain technology has emerged as a promising solution to address such trust issues by offering a decentralized and transparent data management system. A study investigated the blockchain-based image encryption scheme where the encrypted image data is purely stored in blockchain increasing both the confidentiality and integrity[3].

Not only this the healthcare sector also generates a diverse form of medical data which include scanned images, computerized patient records and confidential information creating challenges for researchers who seek to protect this sensitive information. Traditional methods for storing and transmitting the data, especially through the public cloud environment, expose such data to risks such as eavesdropping and data breaches. Previous works have demonstrated that encrypting medical images before transmission can mitigate these risks with blockchain emerging as a viable solution due to the decentralized structure. A blockchain-based chaotic Arnold's cat map encryption scheme (BCAES) was proposed which utilized the Arnold cat map for image encryption before sending the encrypted data to the cloud while simultaneously storing a signed document of the plain image on the blockchain. This combination of cloud and blockchain ensures integrity authenticity and confidentiality[6].

Another color image encryption scheme utilized the Hennon-zigzag map and chaotic restricted Boltzmann machine (CRBM), which significantly improved security by using a two-phased process of permutation and diffusion. In the proposed scheme the permutation phase employed a henna zigzag map to modulate pseudo-random number sequences for row and column permutations which enhanced the scrambling process. In the diffusion phase, the CRBM generated three separate pseudo-random number sequences which were XORed with the RGB channels of the image separately ensuring the encryption throughout all three channels. This combination of advanced encryption algorithms underscores the growing trend of using decentralized networks for verifying data authenticity[17].

A novel blockchain enables secure optimal lightweight cryptography-based encryption BC-LWCIE technique was proposed for the industry 4.0 environment. This proposed scheme integrates the optimal lightweight cryptography LWC, hash functions and optimal key generation by using the chicken swarm optimization (CSO) algorithm that enhances the security by maximizing the Peak Signal-to-Noise Ratio (PSNR). Additionally, the cryptographic pixel values of the deciphered images are securely stored on the blockchain. This technique has outperformed the recent encryption techniques which highlights its effectiveness in securing the image data in IoT environment[18].

Contributions of paper

The contributions of this paper are as follows:

1. Introducing a new hybrid image encryption method combining the chaotic maps (logistic and May map) and fractal-based permutation techniques to achieve robust security for breast cancer images.
2. Employing the May map to generate a chaotic sequence and computing an S-box from it introducing a high level of randomness and unpredictability in the encryption process
3. Proposed the integration of a fractal generation method to further shuffle the image pixels after chaotic permutation which adds an extra layer of complexity to the encryption
4. Demonstrates the resilience of the proposed encryption scheme against common cryptographic attacks including differential attacks statistical attacks and brute force attacks

The structure of this paper is as follows: section "Fundamental theory" discuss the fundamental theory of the maps used in the proposed scheme, section "Proposed blockchain-orchestrated multi-cloud" explains the blockchain orchestrated multi-cloud environment and its working, section "Proposed image encryption process" contains the proposed image encryption and decryption technique, section "Simulation results and security analysis" contains the simulations and results and section "Discussion" and "conclusion" concludes the paper.

Fundamental theory

Following fundamental theories are used in our proposed algorithm:

May map

The May Map is a 1D chaotic map which was named after Robert M. May who introduced it as a simple model to study the population dynamics. The May map is characterized by its ability to generate chaotic sequences which depend on the value of its parameters. The standard equation for the May Map is given by:

$$x_{n+1} = x_n.exp(r.(1 - x_n) \tag{1}$$

Here, x_n is the current state of the system and x_0 is the initial value. r is the control parameter which determines the chaotic behaviour of the map and x_{n+1} is the next state in the sequence. The May Map exhibits chaotic behaviour when the control parameter r is within a certain range, typically around $r > 2.5$. As r increases, the system undergoes bifurcation which leads to a complex dynamics and chaos[19].

The bifurcation diagram of May map shows the transition from stability to chaos as the parameter r increases. A single stable point splits into two then four and then so on. This is known as period doubling which lead to complex behaviour.

The Lyapunov diagram of May map provides a visual representation of the system's behaviour as r varies. When $\lambda < 0$ the system shows a stable and non-chaotic behaviour. This region is typically appear as dips or valleys below the zero line as shown in Fig. 1. When $\lambda > 0$ the system shows a chaotic behavior as the positive exponent indicates that trajectories starting from nearby initial conditions diverge exponentially that ultimately leads to unpredictable behaviour.

Logistic map

The logistic map is a fundamental mathematical model in the study of chaotic systems. Despite its simplicity, logistic map shows complex and chaotic behavior under specific conditions making it an important part of chaotic theory. The logistic map is defined mathematically as:

$$x_{n+1} = r.x_n.(1 - x_n) \tag{2}$$

Here, x_n is the population at iteration n, chosen normally between 0 and 1 and r is the growth rate parameter typically $r \in [0,4]$.

The behaviour of logistic map changes drastically with different values of the growth rate r. At $r > 3.57$ the system enters the chaotic regime where the behaviour is highly sensitive to initial conditions and at $r = 4$, the system exhibits full chaos[20]. One of the most important aspects of logistic map is its bifurcation diagram which represents the transition from order to chaos as the values of parameter r is changed. The Fig. 2 shows the bifurcation diagram of logistic map representing that how a stable point can split into two, four and eight points and eventually leading to a chaotic state.

Sierpinksi triangle fractal pattern

The Sierpinksi triangle, also known as the Sierpinksi gasket is a well known fractal pattern named after the Polish mathematician Waclaw Sierpinksi. It is self-similar structure and a classic example of geometric shape with a fractional dimension falling between 1D lines and 2D surfaces. The Sierpinksi triangle is constructed by using an iterative process where we start with an equilateral solid triangle and subdivide it into four smaller congruent equilateral triangles and remove the central one. For the remaining triangles, same subdivision and removal is repeated[21]. Continuing this leads to the fractal pattern called Sierpinksi triangle. The fractal dimension D of the Sierpinksi triangle is given by Eq. 3,

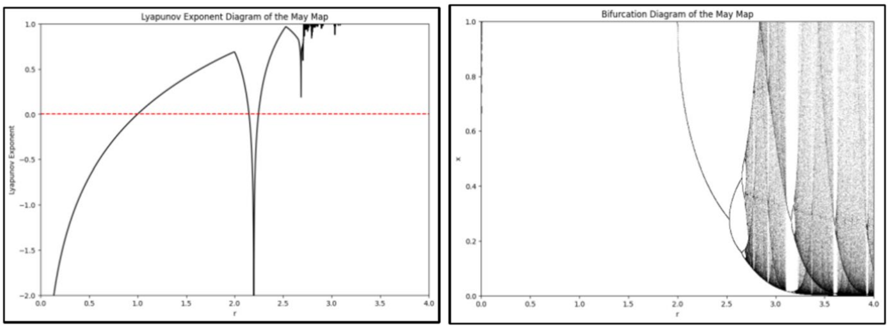


Fig. 1. Bifurcation and Lyapunov exponent diagram of may map.

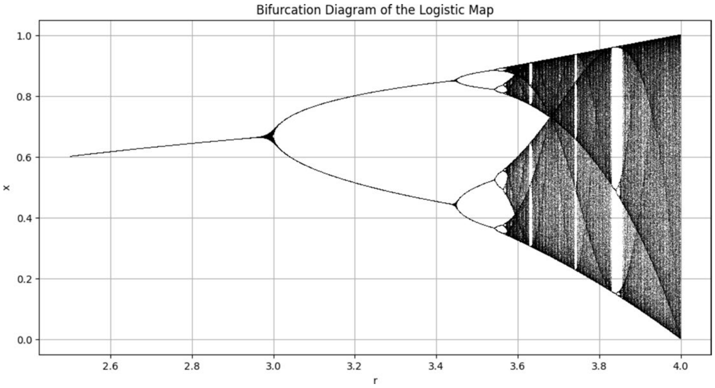


Fig. 2. Bifurcation diagram of logistic map.

$$D = \frac{log3}{log2} \approx 1.585 \tag{3}$$

It shows us that the triangle is complex since the dimension is non-integer.

The Sierpinksi triangle is a fractal pattern that is used in image encryption due to its inherent properties of self-similarity and complexity. It can be used to design robust encryption schemes. The fractal pattern of Sierpinksi triangle such as shown in Fig. 3 is used for generating complex key that are sensitive to initial conditions. The recursive nature of these triangles ensure that a small change in the key gives an entire different encrypted image. The Sierpinksi triangle can also be used for permutation and diffusion process as mapping image according to the Sierpinksi pattern can make the original structure disrupted to such an order that it cannot be reversed by unauthorized party[22].

Chebyshev map

The Chebyshev Map generates chaotic sequence based on the following equation

$$x_{n+1} = cos(k.arccos(x_n))$$

Here, k is the control parameter, usually $k \geq 2$. x_n is the value at iteration n with an initial value x_o.

Proposed Blockchain-Orchestrated multi-cloud

Our proposed algorithm leverages the IoT devices for data capture and employs blockchain orchestrated multi-cloud environment, May Map, Logistic Map and Fractal Based Image encryption scheme to secure the Breast cancer images in hospital settings. The subsequent steps outline the work flow from image capture to decryption and assess by the end user.

Image capture via IoT devices

The initial stage of the process involves the capture of images through IoT enabled devices. These devices may include cameras, sensors or other imaging tools which are deployed within the operational environment to correct a real time image data. The IOT devices continuously monitor and capture the images in a hospital. Before the captured image is transmitted, preliminary processes like compression can be performed on the devices for optimizing the data for secure transmission.

Image encryption process

After receiving the data it is encrypted by using the encryption scheme. Moreover, digital signatures using the hash function are also generated for the images. This dual approach safeguards both the integrity and confidentiality of the data ensuring that the image remains unaltered and accessible only to the authorized recipients. By using these security measures the sender effectively protects the image from tempering and unauthorized access during storage or transmission.

Blockchain-Orchestrated management

After the encryption of images the blockchain orchestrator functions as a decentralized control hub that ensures the secure and efficient handling of data. It is responsible to allocate the storage resources across various cloud providers based on the criteria such as performance, cost efficiency and redundancy requirements. The encrypted

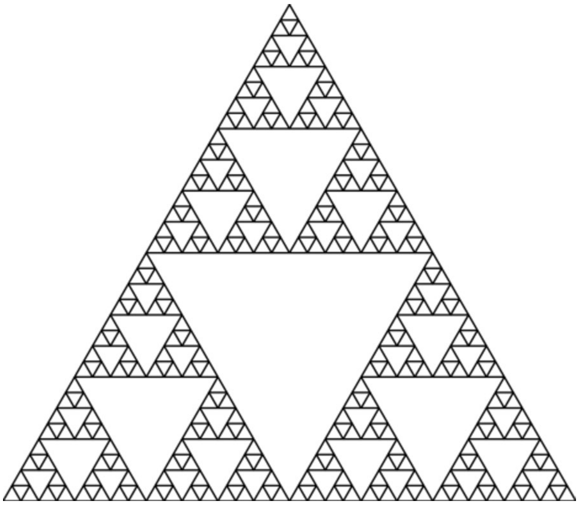


Fig. 3. Sierpinksi triangle.

images along with their associated Meta data are distributed across the selected cloud providers. The blockchain maintains a transparent record of where each data is stored to ensure traceability and integrity.

Cloud storage and verification

Upon distribution to the cloud providers the encrypted images are securely stored within the respective cloud environment. The encrypted images are replicated across multiple cloud providers which enhances the data availability and provides a fail-safe mechanism in case of data corruption or loss at any single provider. The blockchain continuously verifies the integrity of the stored images by comparing their hashes against the original hash functions recorded in the blockchain. Any changes detected triggers the immediate alert thereby safeguarding data against unauthorized modification or tampering.

Image decryption and access

When an authorized user request access to the stored images the decryption process is initiated. This step involves the retrieval of encrypted images and their associated decryption key followed by the decryption of the images for end-user access. This process can be carried out either within the cloud environment or locally depending on the specific use and security requirements. After the images are decrypted and available to the end user this step completes the cycle ensuring the user disease images which are both secure and unaltered from their original form.

The proposed system, illustrated in Fig. 4, is designed to continuously monitor all the activities related to image encryption storage and access. Regular audits are conducted which ensures that the system remains robust against evolving cyber security challenges. By combining the strengths of IOT blockchain and cloud computing the system ensures confidentiality integrity and availability of sensitive image data thereby addressing the critical security concerns in data management.

Our proposed hybrid image encryption method combines the chaotic maps with fractal-based permutation techniques offering some significant advantages over traditional centralized cloud models. By leveraging decentralized processing, our approach reduces the reliance on centralized servers, thereby decreasing associated infrastructure and maintenance costs. Additionally, this decentralized framework not only improves computational efficiency by distributing processing tasks but also minimizes delays and avoids system failure which is common in centralized setups.

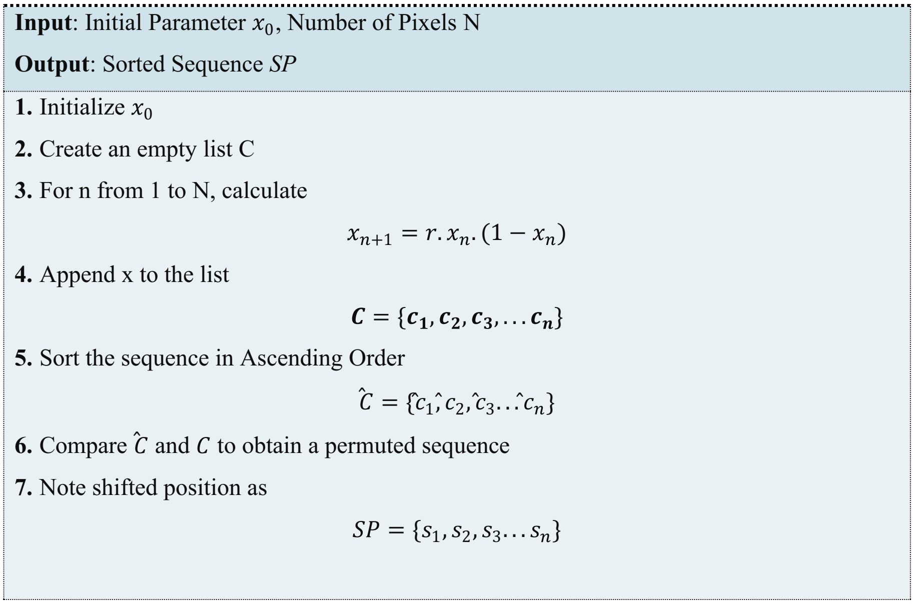
Proposed image encryption process

The details of the proposed image encryption algorithm is given below:

Encryption process

Step 1: Chaotic sequence generation using logistic map

A chaotic sequence is generated using the logistic map which can be shown in Fig. 7. The logistic map is iteratively applied to produce a sequence of chaotic values. The initial value x_0 and the total number of iterations N are chosen based on the number of pixels of image. For example, for an image of size 256 × 256, $N = 65536$. Once the chaotic sequence is generated, it is sorted in ascending order and compared with the original sequence. The change in positions is noted as *SP*. This shifted position sequence serves as the basis for permuting the pixel positions in the image thereby introducing non-linearity and diffusion into the encryption process.



```
Input: Initial Parameter $x_0$, Number of Pixels N
Output: Sorted Sequence $SP$
1. Initialize $x_0$
2. Create an empty list C
3. For n from 1 to N, calculate
        $x_{n+1} = r.x_n.(1 - x_n)$
4. Append x to the list
        $\boldsymbol{C = \{c_1, c_2, c_3, \ldots c_n\}}$
5. Sort the sequence in Ascending Order
        $\hat{C} = \{c_1\hat{,}c_2,\hat{}c_3\ldots\hat{}c_n\}$
6. Compare $\hat{C}$ and $C$ to obtain a permuted sequence
7. Note shifted position as
        $SP = \{s_1, s_2, s_3\ldots s_n\}$
```

Algorithm 1. Sorted sequence algorithm.

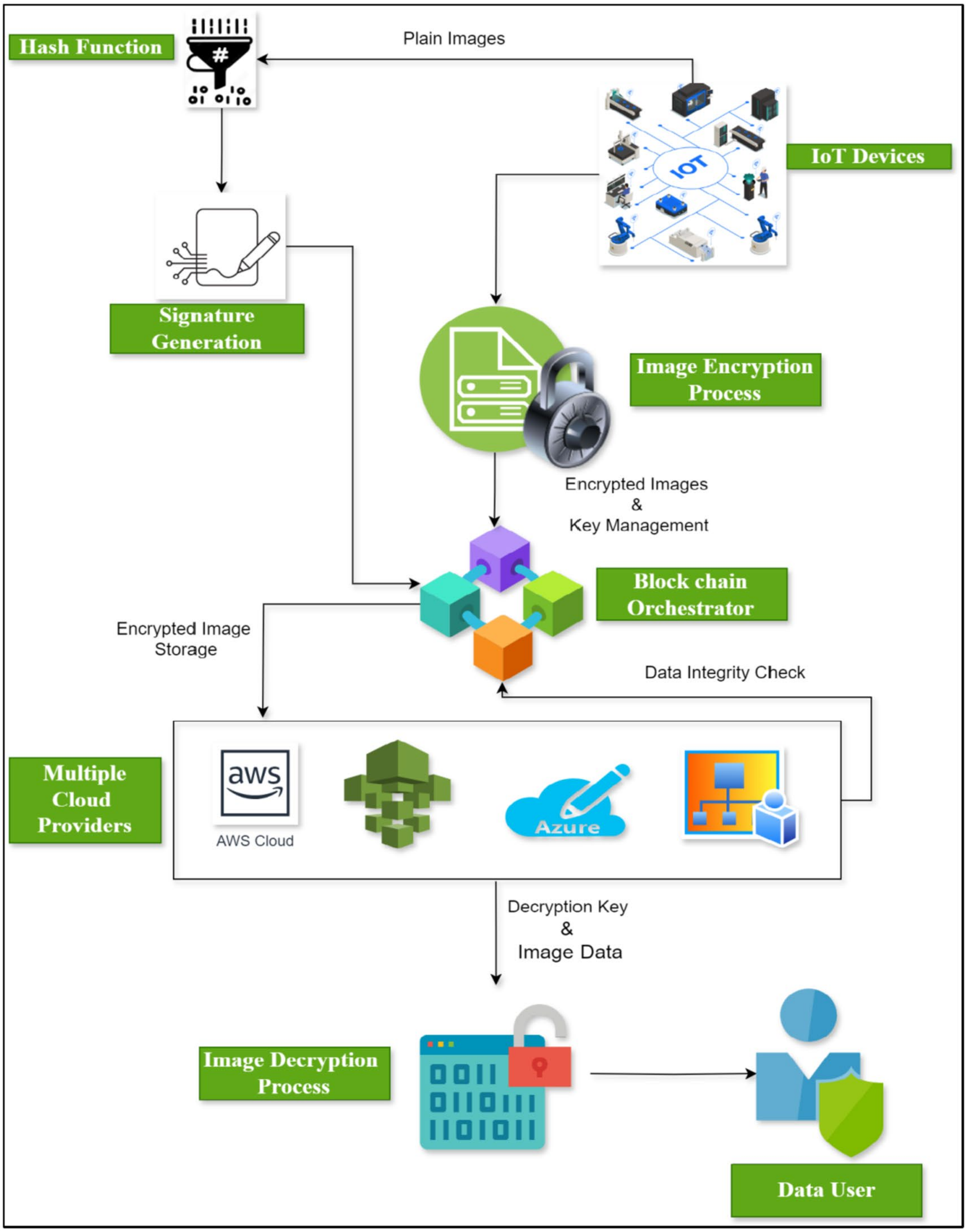
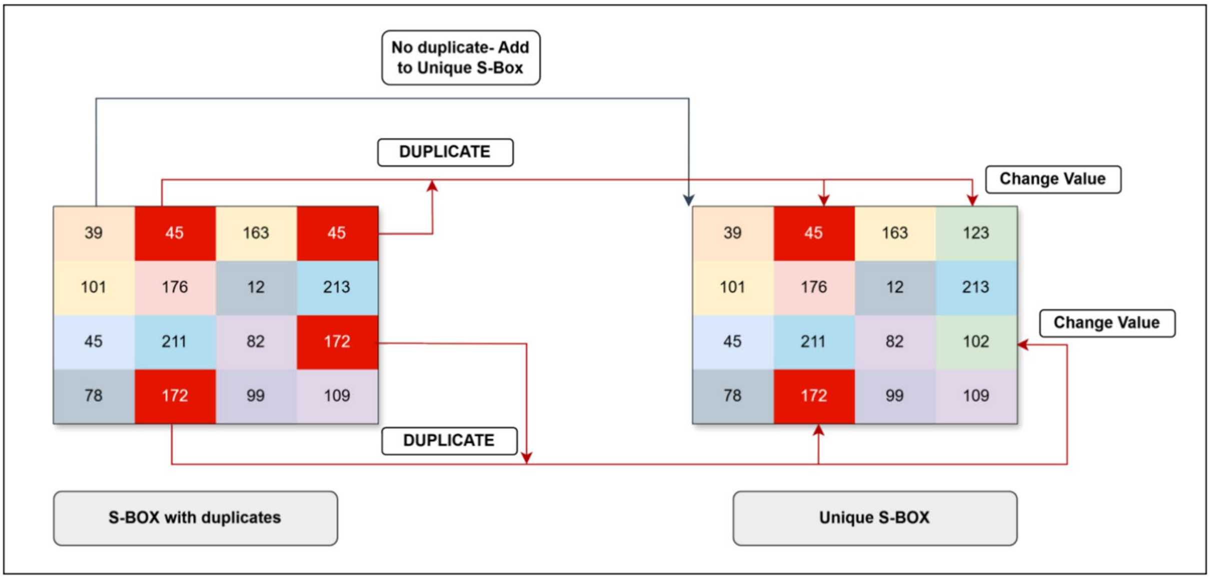


Fig. 4. Primary components in proposed Blockchain-Orchestrated multi-cloud.

Step 2: S-box generation using may map
In the first step will initialize the May maps parameter by setting the initial value and the number of iterations to be performed. The chaotic sequence is generated by iterating the May map where each value is calculated by using the formula in Eq. (1). The sequence of the values is sorted in an array S. Subsequently the values in S are normalized to fit within a 8-bit range (0 to 255) that ensures that the resulting S-box entries are suitable for cryptographic operations. The normalized values are then used to generate an initial S-box where each position in the S-box is filled by the values derived from the chaotic sequence. This ensures the high degree of randomness and unpredictability to enhance the uniqueness of S-box. The duplicate values are identified and adjusted.

0	**1**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
17	20	23	24	30	32	37	38	39	40	47	54	58	61	62	63
64	65	66	67	68	69	78	79	81	83	85	87	88	96	97	104
105	110	112	113	114	115	122	124	125	127	129	131	133	138	144	146
147	148	154	156	157	165	166	173	177	178	179	180	181	182	185	186
187	188	189	190	191	192	193	194	195	196	197	198	199	202	206	207
209	210	217	218	219	220	223	224	227	228	236	247	248	249	250	251
254	255	78	184	22	148	99	68	9	234	128	9	146	190	80	73
226	20	164	104	127	47	244	25	175	194	190	205	53	159	232	11
31	246	60	211	80	252	75	196	169	227	160	139	36	37	58	116
9	159	223	173	39	140	202	94	251	214	250	194	38	102	53	52
116	212	153	226	89	32	145	246	182	147	164	87	57	103	160	181
208	35	161	141	156	98	35	203	105	28	252	248	154	246	227	255
111	57	214	203	151	144	48	33	167	32	1	43	230	113	99	37
147	152	36	131	32	43	252	82	253	6	6	97	48	64	191	78
11	225	45	197	218	131	188	226	155	150	178	52	172	234	137	2

Table 1. S-Box computed by Algorithm 1.

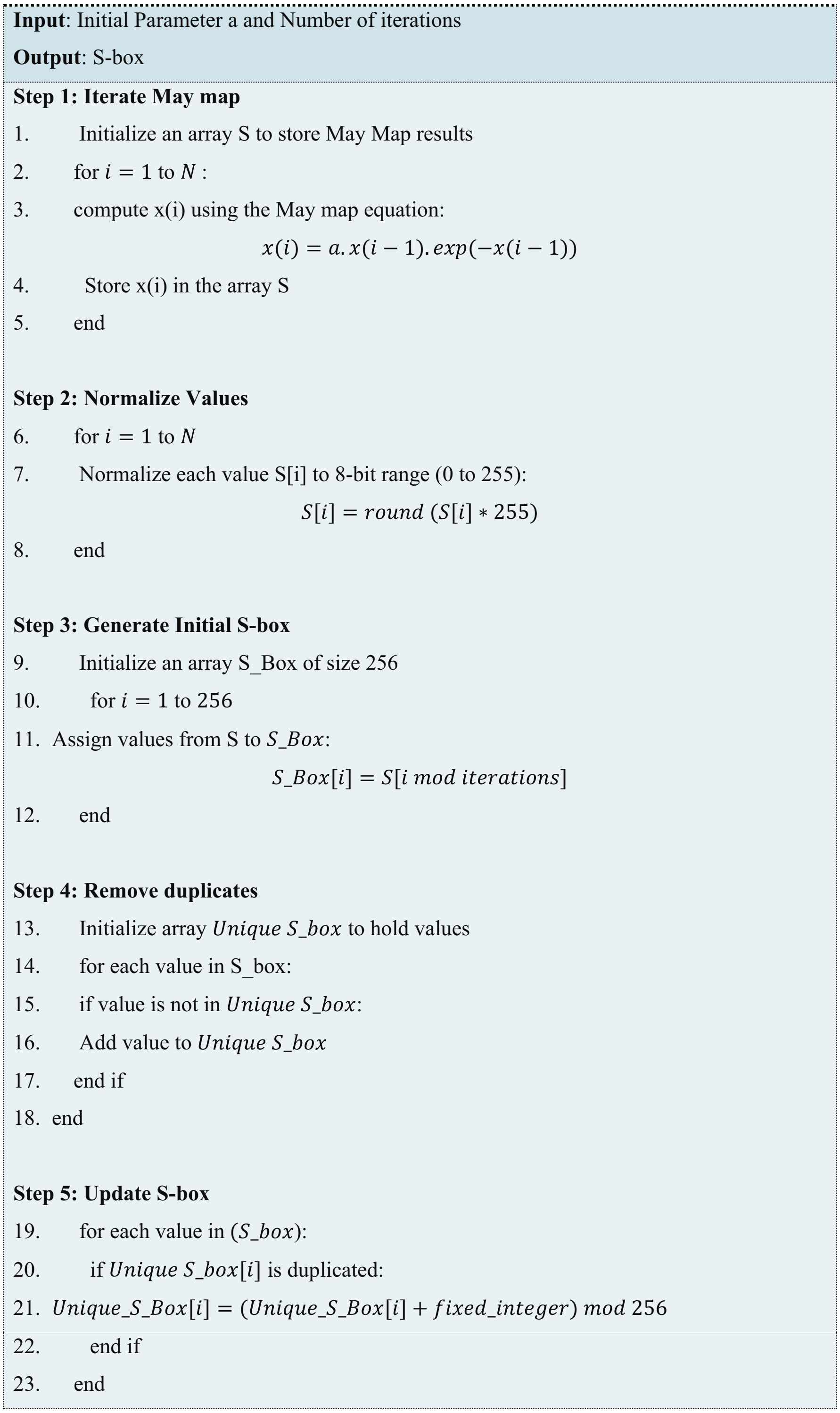


39	45	163	45
101	176	12	213
45	211	82	172
78	172	99	109

39	45	163	123
101	176	12	213
45	211	82	102
78	172	99	109

Fig. 5. Visualization of duplicate removal process in S-box generation.

This is done by adding a fixed integer to each duplicate value followed by a modulo operation so that the result remains within the eight bit range. This step is important for maintaining the bijective property of the S-box. Finally the resulting S-box is ready to use in the secure image encryption scheme. The Algorithm 1 gives the pseudo code for S-box generation. Table 1 provides an illustration of proposed S-box and Fig. 5 visualizes the duplicate removal process in S-Box generation.



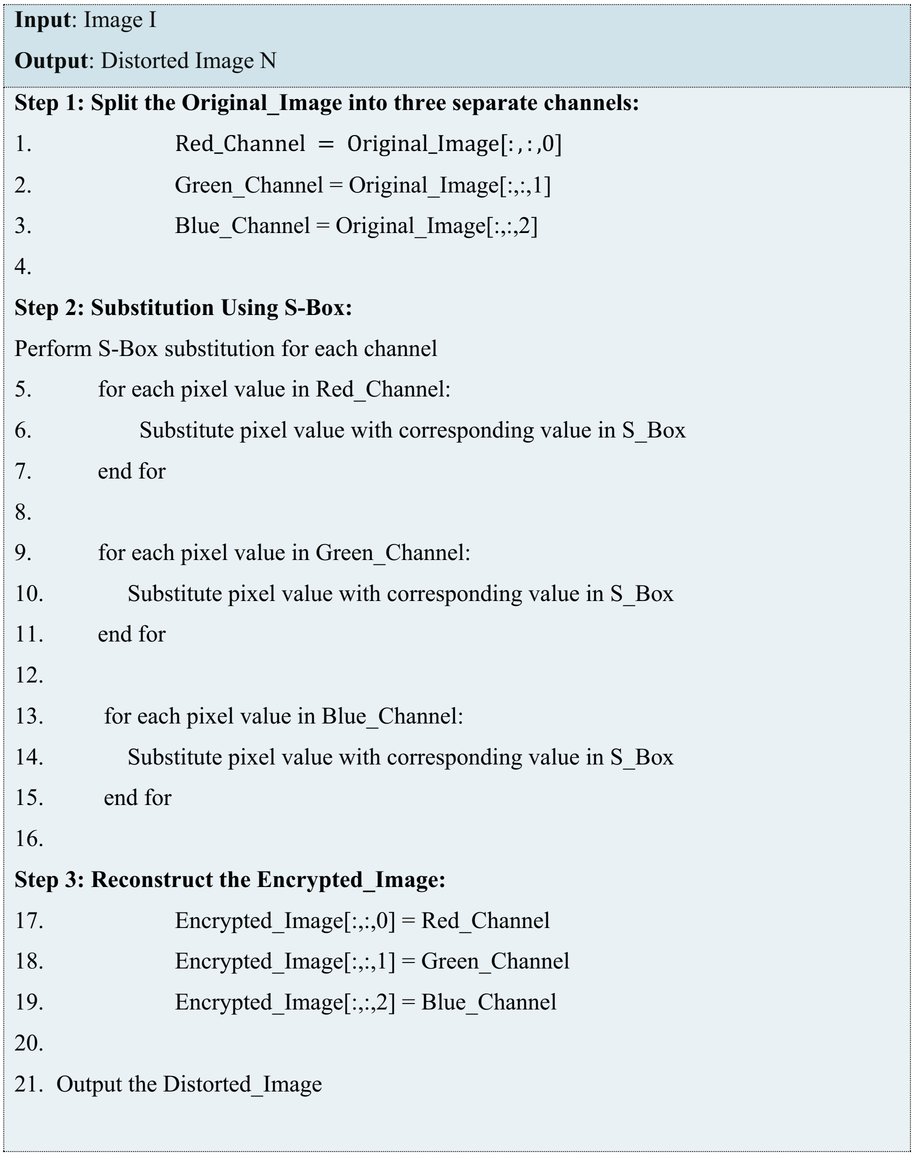
Input: Initial Parameter a and Number of iterations

Output: S-box

```
Step 1: Iterate May map
    Initialize an array S to store May Map results
    for $i = 1$ to $N$ :
        compute x(i) using the May map equation:
            $x(i) = a.x(i-1).exp(-x(i-1))$
        Store x(i) in the array S
    end

Step 2: Normalize Values
    for $i = 1$ to $N$
        Normalize each value S[i] to 8-bit range (0 to 255):
            $S[i] = round\ (S[i] * 255)$
    end

Step 3: Generate Initial S-box
    Initialize an array S_Box of size 256
    for $i = 1$ to $256$
        Assign values from S to $S\_Box$:
            $S\_Box[i] = S[i\ mod\ iterations]$
    end

Step 4: Remove duplicates
    Initialize array $Unique\ S\_box$ to hold values
    for each value in S_box:
        if value is not in $Unique\ S\_box$:
            Add value to $Unique\ S\_box$
        end if
    end

Step 5: Update S-box
    for each value in ($S\_box$):
        if $Unique\ S\_box[i]$ is duplicated:
            $Unique\_S\_Box[i] = (Unique\_S\_Box[i] + fixed\_integer)\ mod\ 256$
        end if
    end
```

Algorithm 2. S-box generation algorithm.

Step 3: Applying S-box to each channel of original image
The original image *M* is first split into its three color channels Red, Green and Blue. Each channel represents its intensity values for its respective color across all pixels in the image. The image pixels are substituted by applying S-Box to each channel separately. This substitution process introduces non-linearity and enhances the security of the encryption process by increasing confusion and diffusion of image data. After substitution, the channels recombine to form a distorted image N.



```
Input: Image I
Output: Distorted Image N

Step 1: Split the Original_Image into three separate channels:
    Red_Channel = Original_Image[:,:,0]
    Green_Channel = Original_Image[:,:,1]
    Blue_Channel = Original_Image[:,:,2]

Step 2: Substitution Using S-Box:
Perform S-Box substitution for each channel
    for each pixel value in Red_Channel:
        Substitute pixel value with corresponding value in S_Box
    end for

    for each pixel value in Green_Channel:
        Substitute pixel value with corresponding value in S_Box
    end for

    for each pixel value in Blue_Channel:
        Substitute pixel value with corresponding value in S_Box
    end for

Step 3: Reconstruct the Encrypted_Image:
    Encrypted_Image[:,:,0] = Red_Channel
    Encrypted_Image[:,:,1] = Green_Channel
    Encrypted_Image[:,:,2] = Blue_Channel

Output the Distorted_Image
```

Algorithm 3. Distorted image algorithm.

Step 4: Diffusion using Chebyshev map

The diffusion step uses the Chebyshev Map, a chaotic system, to generate a sequence of pseudo random numbers from an initial value x_o and a chaos control parameter k. These numbers are normalized to produce indices corresponding to pixel positions in the image. The distorted image pixel values from step 3 are rearranged (permuted) based on these indices, effectively scrambling the image. This process ensures that a slight alteration in the initial conditions or the key results in a completely different permutation.

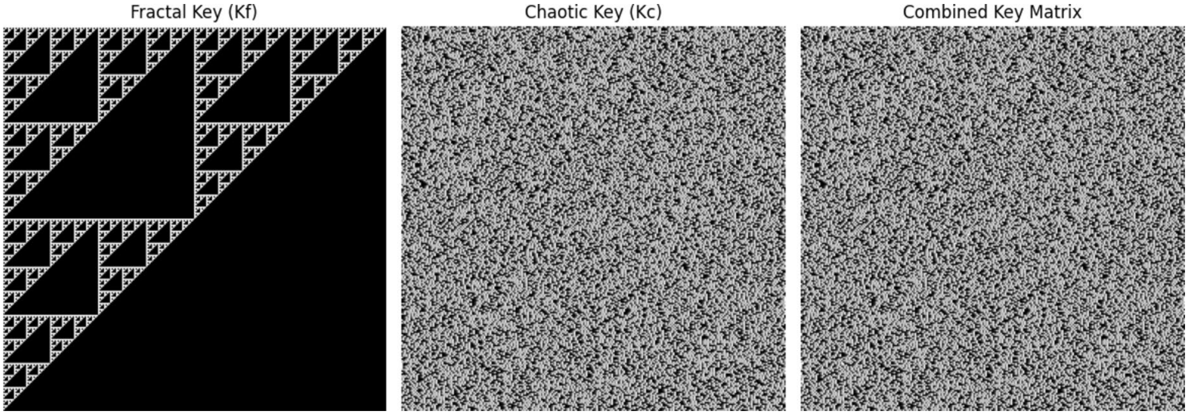
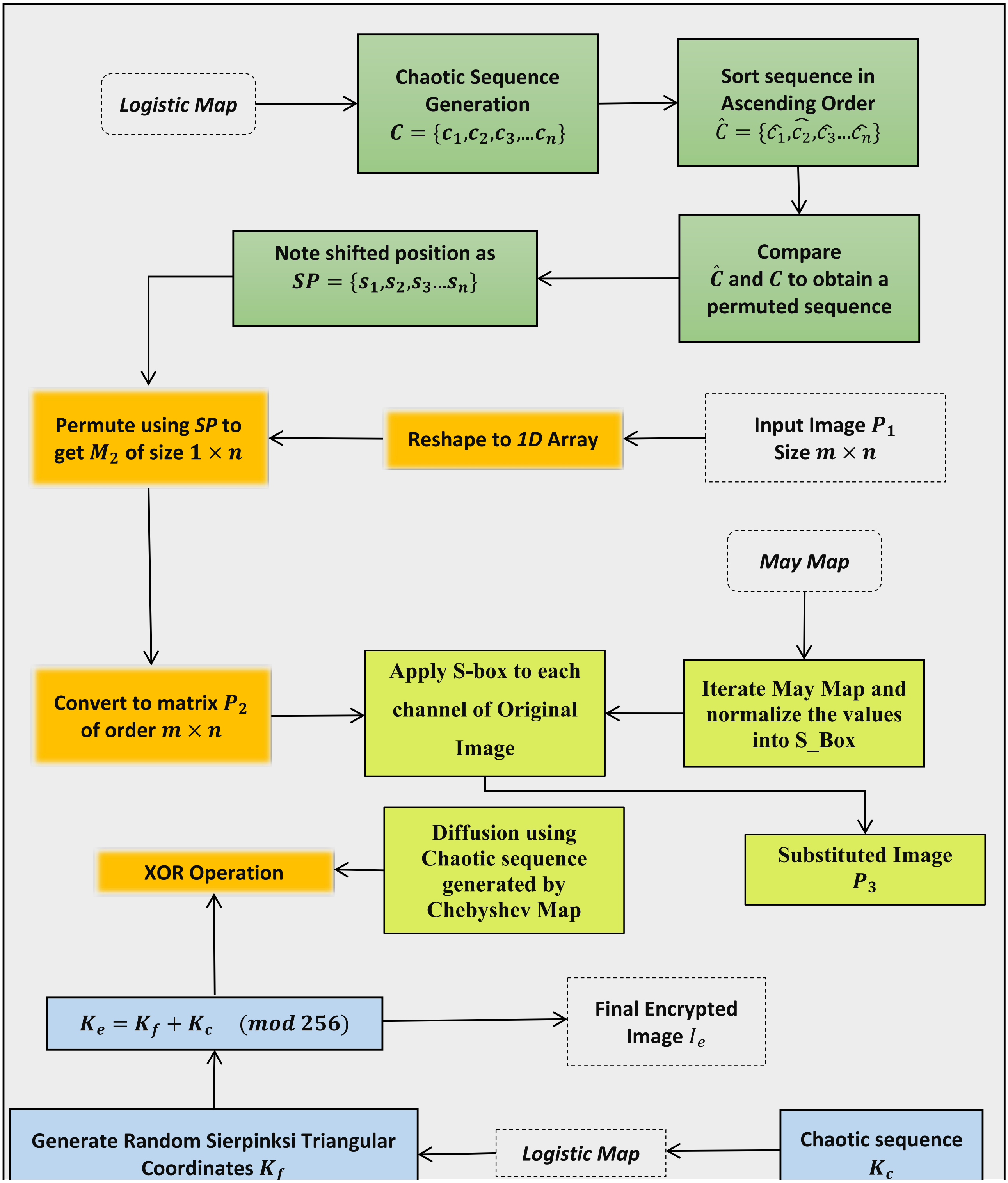


Fig. 6. Fractal-key generation.

```
Input: Distorted Image N, Initial value $x_o$, Control Parameter $k$
Output: Diffused Image D

Step 1: Flatten the distorted image into 1D Array
    $Flatten\_Image \leftarrow Flatten\ (N)$
Step 2: Generate a chaotic sequence using the Chebyshev Map:
    $N = Length\ (Flattened\_Image)$
    $Chaotic\_Sequence = []$
    $x_{n+1} = x_o$
    For i = 1 to N:
        $x_{n+1} = cos(k.arccos(x_n))$
        Append $x$ to $Chaotic\_Sequence$
    End for

Step 3: Normalize Chaotic Values to Generate Permutation Indices:
    Indices=[]
    For each value x in $Chaotic\_Sequence$:
        Index=round(((x+1)/2)*(N-1)+1)
        Append index to indices
    End for

Step 4: Permute the pixel Values based on indices
    For i=1 to N:
        Permuted_Image[Indices [i]-1]= Flattened_Image [i]
    End for

Step 5: Reshape Permuted Image back to original Dimensions
    Diffused_Image=Reshape(Permuted_Image,Dimensions(I))
    Output Diffused_Image
```

Algorithm 4. Diffusion algorithm.



Logistic Map
Chaotic Sequence Generation $C = \{c_1,c_2,c_3,\ldots c_n\}$
Sort sequence in Ascending Order $\hat{C} = \{\widehat{c_1},\widehat{c_2},\widehat{c_3}\ldots\widehat{c_n}\}$
Compare $\hat{C}$ and C to obtain a permuted sequence
Note shifted position as $SP = \{s_1,s_2,s_3\ldots s_n\}$
Permute using SP to get M_2 of size $1 \times n$
Reshape to $1D$ Array
Input Image P_1 Size $m \times n$
May Map
Convert to matrix P_2 of order $m \times n$
Apply S-box to each channel of Original Image
Iterate May Map and normalize the values into S_Box
Substituted Image P_3
XOR Operation
Diffusion using Chaotic sequence generated by Chebyshev Map
$K_e = K_f + K_c \quad (mod\ 256)$
Final Encrypted Image I_e
Generate Random Sierpinksi Triangular Coordinates K_f
Logistic Map
Chaotic sequence K_c

Fig. 7. Flowchart of encryption process.

Step 5: Fractal-based encryption key generation

A Sierpinski Triangle fractal pattern is used as the basis for encryption key generation. The fractal is generated using the logistic map defined in Eq. 2.

Here $r = 3.999$ is the control parameter for chaotic behaviour and x_0 is the initial value. The logistic map is iterated to produce the values of Sierpinski triangle within a 256×256 matrix. The resulting binary fractal pattern where the non-zero values are set to 255, form the initial encryption key matrix K_f. This key is further combined with the chaotic sequence generated from the logistic map. A secondary key matrix K_c is generated by iterating the logistic map with each value scaled to the range [0,255]. The final encryption key K_e is obtained by performing the XOR operation to ensure that pixel values remain within the 8-bit range:

$$K_e = K_f + K_c(mod256) \tag{4}$$

To elucidate the role of Sierpinski triangle in the key generation process, we have presented a visual representations of fractal pattern used in Fig. 6. The Sierpinski triangle, a well-known fractal, is constructed through recursive subdivision of an equilateral triangles, resulting in a complex, and self-similar structure. In our encryption scheme, these fractal patterns introduce a high degree of complexity and randomness, enhancing the diffusion and confusion properties essential for robust encryption (Fig. 7). The inherent self-similarity and intricate design of the Sierpinski triangle ensures that even minimal alterations in the input leads to significant and unpredictable changes in the encrypted output, thereby substantially strengthening the encryption against potential attacks

```
Input: Parameters $r$, $x_0$
Output: Fractal-based key matrix $K_f$
1. Initialize $r = 3.999$ and $x_0$
2. For each iteration in the pixels of matrix perform:
        $x_{n+1} = r.x_n.(1 - x_n)$
3. Calculate Sierpinksi Triangle coordinates $(s, t)$
4. For each pixel position $(i, j)$ in the matrix:
        If coordinates $(i, j)$ match any $(s, t)$ from the Sierpinski Triangle set, set
            $K_f[i, j] = 255\ (white)$
        Otherwise, set to $K_f[i, j] = 0(black)$
5. Store the resulting pattern as $K_f$
```

Algorithm 5. Generation of fractal-based key algorithm.

```
Input: Fractal-based Key Matrix $K_f$
Output: Final Encryption Key Matrix $K_e$
1. Generate a chaotic sequence $K_c$ using the logistic map
2. Initialize $r$ and $x_0$
3. Scale $x_{n+1}$ to range [0,255]
4. Store the value in $K_c$
5. Combine $K_f$ and $K_c$
        $K_e[i, j] = (K_f[i, j] + K_c[i, j]) \quad mod\ 256$
6. Output $K_e$
```

Algorithm 6. Key generation algorithm..

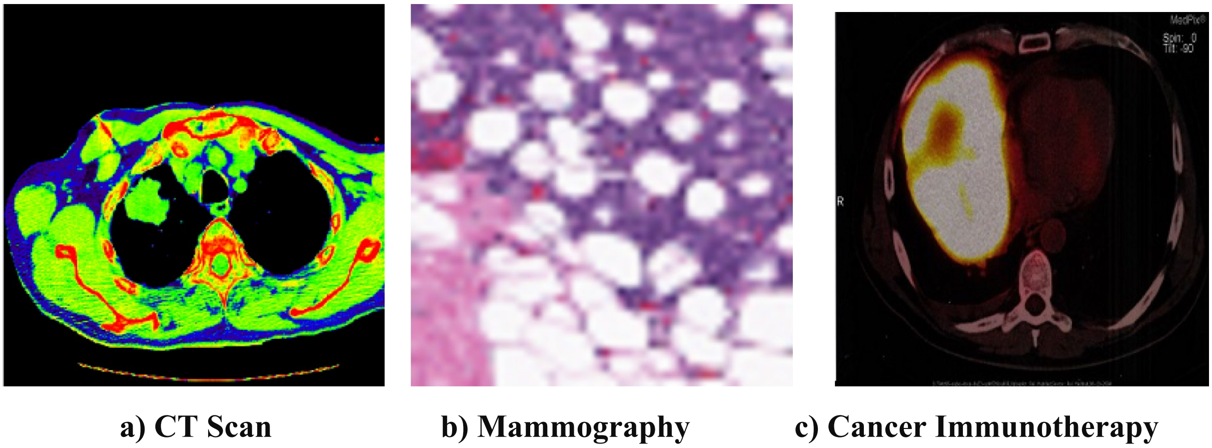
Step 6: XORing

The final encrypted image is generated by performing a pixel-wise XOR operation between the diffused image from step 4 and combined fractal encryption key from step 5 producing the final encrypted image I_e.

$$I_e = D \oplus K_e \tag{5}$$

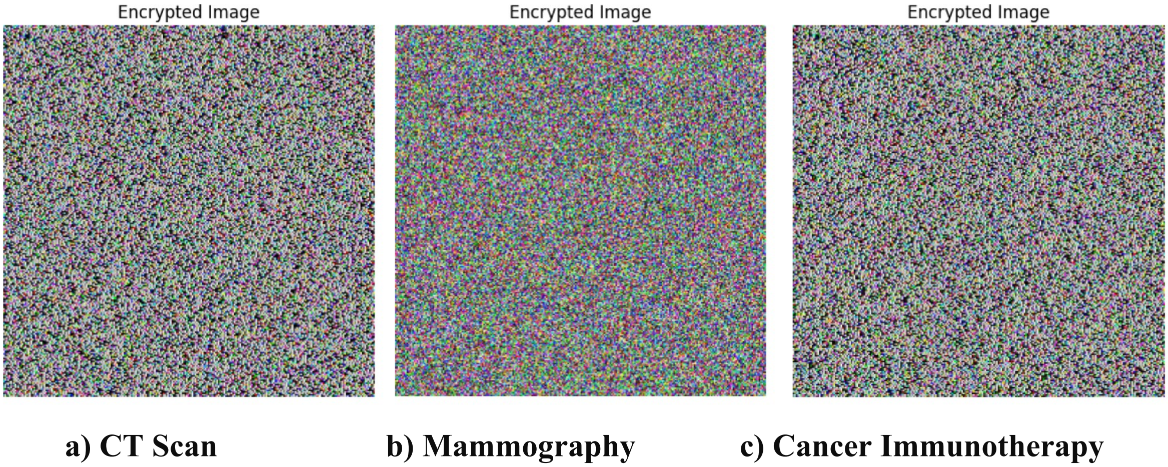
Component	Specifications
CPU	Single-core hyper-threaded Xeon processor at 2.3 GHz
RAM	Approximately 12.6 GB available
Disk space	Around 33 GB available
GPU	Access to GPUs such as NVIDIA Tesla and P100

Table 2. Specifications of development environment Google Colab.



a) CT Scan b) Mammography c) Cancer Immunotherapy

Fig. 8. Original images.



a) CT Scan b) Mammography c) Cancer Immunotherapy

Fig. 9. Encrypted images.

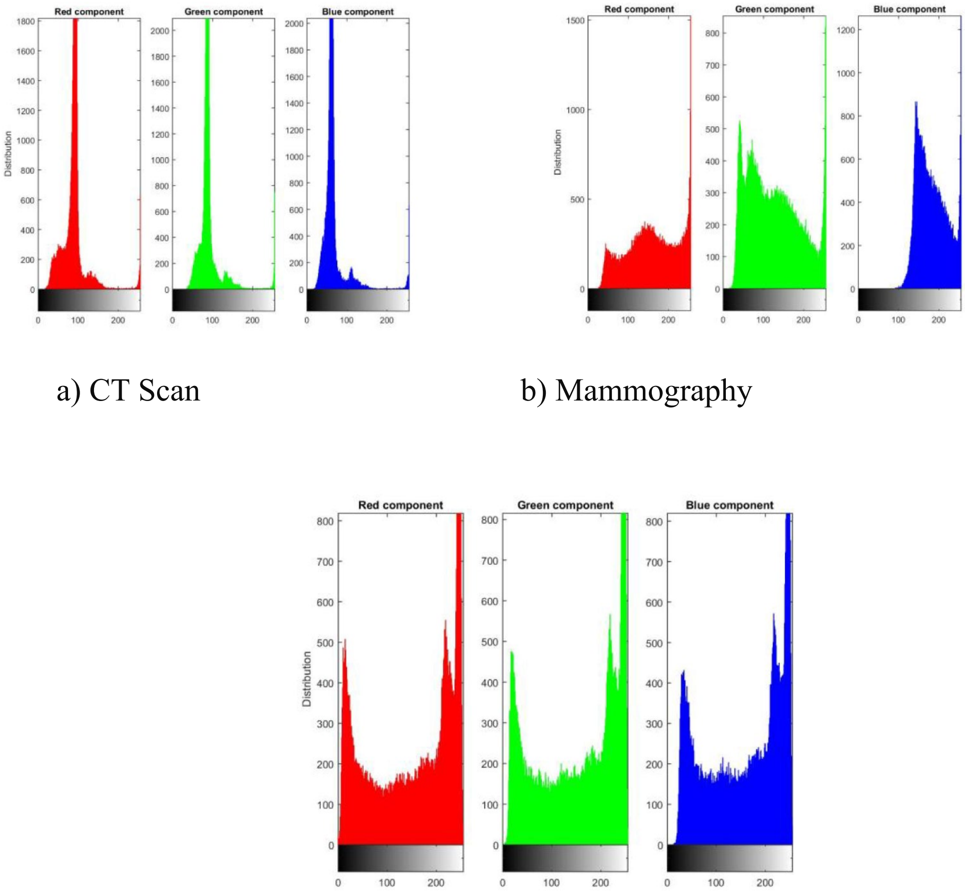
```
Input: Diffused Image $D$ and Key Matrix $K_e[i,j]$
Output: Encrypted Image $I_e$
1. For each pixel $(i,j)$ in $D$:
        $I_e[i,j] = D[i,j] \oplus K_e[i,j]$
2. Output $I_e$
```

Algorithm 7. Encryption algorithm.

Decryption process

The decryption process can be performed as follows:

1. Generate Fractal based key using the same logistic map parameters and initial conditions
2. Find the reverse S-box using the same parameters of May map
3. Perform a pixel-wise XOR Operation between the encrypted and regeneration encryption key matrix to obtain intermediate image
4. Generate chaotic sequence using Chebyshev Map same initial conditions x_o and parameter k. Apply reverse permutation using original sequence of indices.
5. Split the image from step 4 into 3 color channels; Red, Blue and Green
6. Apply inverse S-Box mapping for each pixel and restore the original pixel values.
7. Use the sorted indices from the chaotic sequence to reverse permutation and retrieve original image
8. Reshape 1D array back to the image dimension



a) CT Scan

b) Mammography

c) Cancer Immunotherapy

Fig. 10. Histogram of original images.

Simulation results and security analysis

To evaluate the effectiveness and robustness of the proposed image encryption scheme, various experiments and analysis were conducted. The simulation results are crucial in assessing the statistical qualities of encrypted image. A comprehensive set of tests are performed including histogram analysis, correlation coefficient analysis, key sensitivity and differential attacks like NPCR and UACI. Additionally, the metrics such as Means Square Error (MSE), Peak Signal-to-Noise Ratio (PSNR) and Chi-square tests are used to assess the quantitative insights into the encryption quality. The results from these analysis have demonstrated the strength of encryption scheme against potential attacks, highlighting the capability to maintain data integrity and confidentiality.

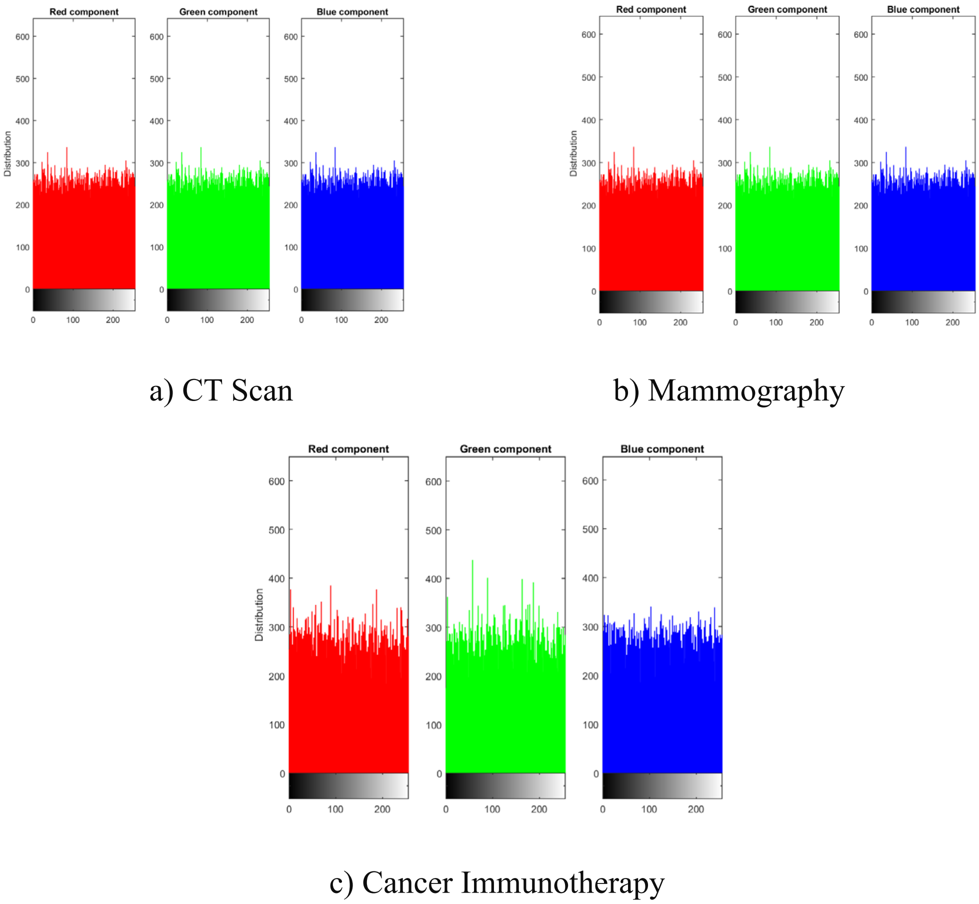
To test the encryption scheme we have used different medical images from datasets. These medical images includes CT scan image[23], Mammography[24] and Cancer immunotherapy[25] shown in Fig. 8. The encrypted images are shown in Fig. 9. For computational experiments we employed Google Colab as our development environment. Google Colab provides access to virtual machines with varying specifications. Table 2 shows specifications.

Histogram analysis

The histogram analysis assess the pixel intensity distribution of the encrypted images compared to plain image. A uniform histogram in the encrypted image indicates that the pixel values are evenly distributed highlighting that the encryption process effectively masks the original content. This uniformity demonstrate that the images are resistant against statistical attacks since no patterns of the image are revealed. The histograms of the test images as shown in Figs. 10 and 11. The visual distribution and randomness in encrypted image indicates high entropy and randomness as illustrated in Fig. 8.

Correlation coefficient analysis

The correlation coefficient analysis measures the relationship between adjacent pixels in both the original and encrypted images. In a secure encryption scheme, the correlation between adjacent pixels should be close to zero which indicates that the encryption process has successfully disrupted the inherent spatial relationships in the original image. On the other hand, a low correlation value indicates that the encrypted image does not



a) CT Scan

b) Mammography

c) Cancer Immunotherapy

Fig. 11. Histogram of encrypted images.

Images	Horizontal direction	Vertical direction
CT scan original	0.0082	0.0019
CT scan encrypted	0.0029	0.001
Mammography original	0.0071	0.0052
Mammography encrypted	0.0010	0.0095
Cancer immunotherapy original	0.0002	0.0050
Cancer immunotherapy encrypted	0.00012	0.00102

Table 3. Correlation values of original and encrypted images along vertical and horizontal direction.

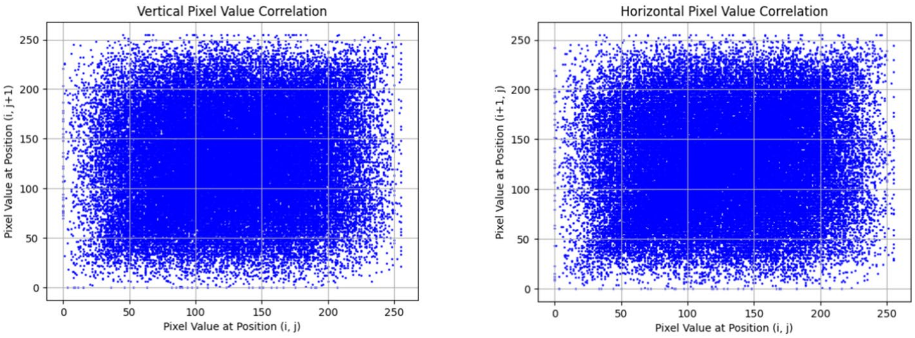


Fig. 12. Vertical and horizontal pixel value correlation of encrypted image of mammography.

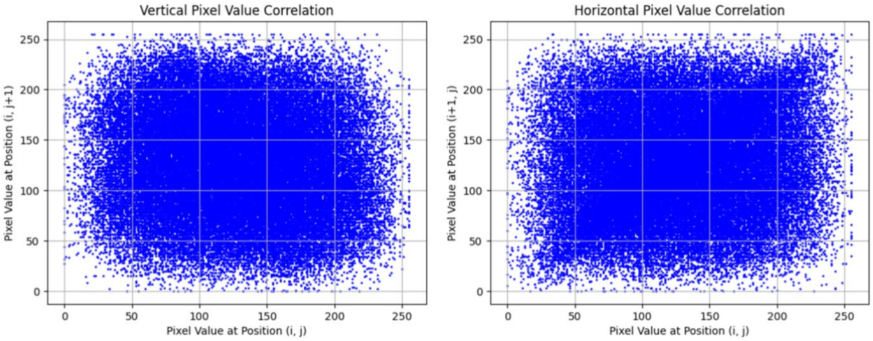


Fig. 13. Vertical and horizontal pixel value correlation of encrypted image of CT scan.

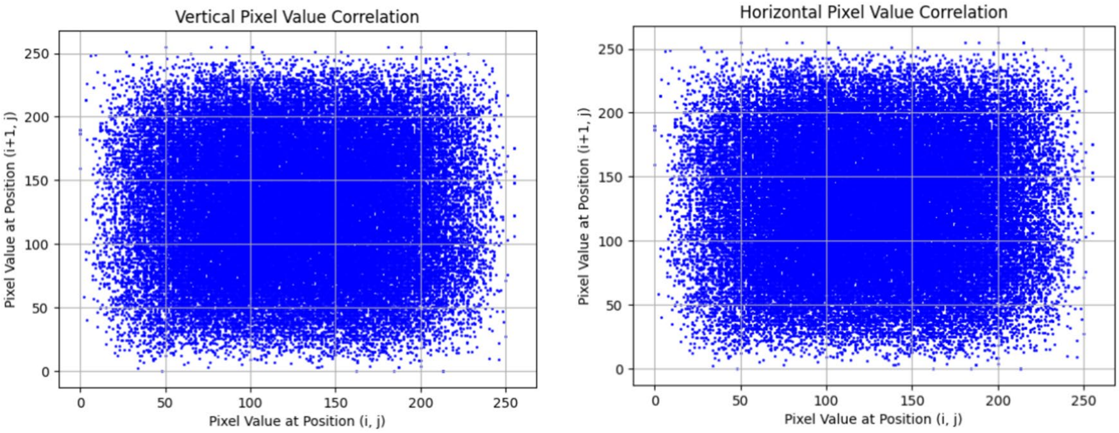


Fig. 14. Vertical and horizontal pixel value correlation of encrypted image of cancer immunotherapy.

retain any significant structural similarities with the plain image which makes it difficult for the intruder to reconstruct the image.

The correlation is measured as follows:

$$Corr(x, y) = \frac{Cov(x, y)}{\sqrt{Var(x) \times Var(y)}} \tag{6}$$

$Cov(x, y)$ is the co-variance between adjacent pixels x and y, $Var(x)$ and $Var(y)$ are the variance values of the pixel x and y respectively. Table 3 indicates the correlation values of original and encrypted images along horizontal and vertical directions. Lower correlation suggests that the encryption algorithm effectively disrupts the relationship between adjacent pixels. Moreover, Figs. 12, 13 and 14 illustrate the relationship between adjacent pixels in the three test images. The scattered distribution of pixel pairs show a lack of correlation between adjacent pixels reflecting the effectiveness of the proposed scheme.

Differential attack analysis

Two metrics, namely Number of Pixel Rate Change (NPCR) and Unified Average Change Intensity (UACI), used to evaluate the resistance of the encryption scheme against differential attacks.

NPCR means the percentage of different pixels between two encrypted images when the original image is subjected to slight modifications such as changing a single pixel. Whereas, UACI quantified the average intensity of the differences between the two encrypted images. Mathematically,

Images	NPCR	UACI	PSNR	MSE	Entropy
CT scan	99.98	33.51	20.10	28,192.47	7.9998
Mammography	99.87	33.35	28.92	87,298.82	7.9981
Cancer immunotherapy	99.92	33.31	29.74	72,721.95	7.9961

Table 4. NPCR, UACI, PSNR and MSE values for test images using proposed scheme.

$$NPCR = \frac{1}{W \times H}\sum_{i=1}^{W}\sum_{j=1}^{H} D\left(i,j\right) \times 100\% \tag{7}$$

$$UACI = \frac{1}{W \times H}\sum_{i=1}^{W}\sum_{j=1}^{H} \frac{\left|C_1(i,j) - C_2(i,j)\right|}{255} \times 100\% \tag{8}$$

W and H are width and height of the image respectively.

$D(i,j)$ is a binary function defined as:

$$D\left(i,j\right) = \begin{cases} 0 \ \ if \ \ C_1\left(i,j\right) = C_2\left(i,j\right) \\ 1 \ \ if \ \ C_1\left(i,j\right) \neq C_2\left(i,j\right) \end{cases} \tag{9}$$

$C_1(i,j), C_2(i,j)$ are the pixel values at position (i,j) of two encrypted images. A value near 99% for NPCR and 33% for UACI indicates that the encryption process can effectively prevent attackers from predicting the effect of minor changes in the plaintext image on the ciphered image which ensures a high level of security.

Encrypted image quality measure

Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR) are the metrics used to assess the visual quality of encrypted image.

Encryption scheme	Entropy value
27	7.9904
28	7.9834
Proposed algorithm	7.998

Table 5. Comparison of entropy values.

Encryption scheme	NPCR	UACI
29	99.603	33.45
30	99.6098	33.4384
Proposed algorithm	99.92	33.31
Encryption scheme	**PSNR**	**MSE**
31	8.440	7764
32	8.4045	8333.3851
Proposed algorithm	29.74	72,721.95

Table 6. Comparison of NPCR and UACI.

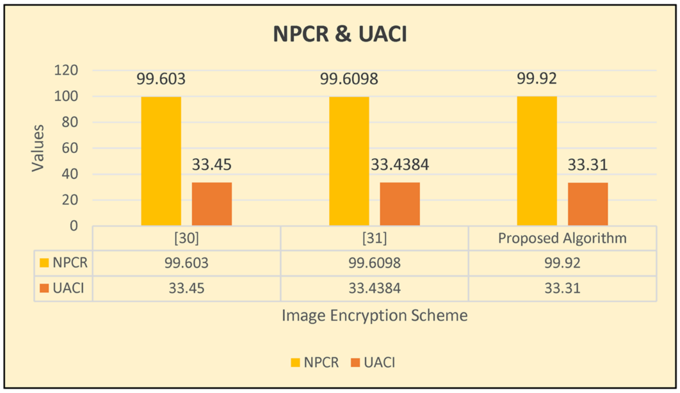


Fig. 15. Graph showing the comparison of entropy values, NPCR & UACI.

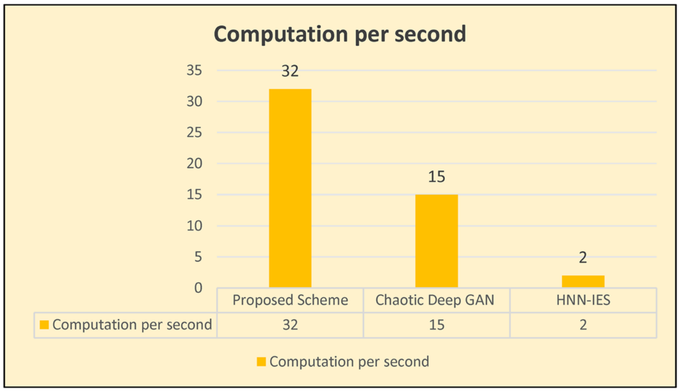


Fig. 16. Time complexity of proposed scheme compared with previous techniques.

MSE calculates the average squared difference between the pixel values of the original and the encrypted images while PSNR measures the ratio of maximum possible pixel value and power of the noise caused by the encryption process. High

$$MSE = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} \left[I\left(i,j\right) - I'\left(i,j\right) \right]^2 \tag{10}$$

W and H are width and height of the image respectively and $I(i,j), I\prime(i,j)$ are the pixel values of the original and encrypted images at position (i,j).

PSNR is given by,

$$PSNR = 10.log_{10} \frac{255^2}{MSE} \tag{11}$$

Here, 255 is the maximum possible pixel value for 8-bit image. MSE values and low PSNR values indicate that the encryption image significantly differs from the original ensuring that the original image's content cannot be reconstructed. In medical imaging, maintaining a high PSNR values are essential because even a minor distortion can lead to misinterpretations, potentially affecting the patient diagnosis and treatment. In Computed tomography (CT) scans, preserving image quality is vital for accurate detection of anomalies. Studies have shown that higher PSNR values means better image quality with clearer details especially in CT images[26].

Entropy analysis

Entropy measures the amount of randomness or unpredictability in an image. It quantifies the amount of information and uncertainty in image pixels. An entropy value near 8 indicates more randomness and suggests that the image is difficult to predict. The entropy H can be calculated as

$$H = -\sum_{i=0}^{N-1} p(x_i) log_2(p(x_i))$$

Here, $p(x_i)$ is the probability of occurrence of pixel value x_i and N is the number of possible pixel values. In Image encryption, Shannon entropy measures the randomness and unpredictability of pixel intensity distributions. Our proposed algorithm achieves an entropy value of 7.9968 closely approaching the ideal 8 indicating maximum randomness and minimal redundancy.

The values of NPCR, UACI, PSNR and MSE shown in Table 4 indicate the effectiveness of encryption scheme. The high NPCR and UACI values indicate substantial pixel and intensity changes between original and encrypted images, crucial for secure encryption. Meanwhile, the MSE values suggests a high level of dissimilarity between the encrypted and original image and high PSNR values indicate that encrypted images have maintained good quality. Table 5 presents the comparison of entropy values with previous techniques and Table 6 and Fig. 15 compares the NPCR and UACI values.

Time complexity

Our proposed scheme can encrypt 32 images of size 256 × 256 per second whereas when compared to existing technique like Chaotic deep GAN and HNN-IES our proposed scheme is significantly faster and efficient as depicted in Fig. 16 [34].

Discussion

In addition to mammography medical images, the proposed scheme can also be applied to other types of medical data such as MRI scans and genetic information. MRI images like CT scans, contain intricate details critical for accurate diagnosis, making the encryption essential to protect patient's privacy. Similarly genetic data which involves sensitive personal information also required a very strong encryption to prevent third-party unauthorized access. The robust security provided by our encryption method can be adapted to secure these data types, further enhancing the potential of the proposed system in safeguarding a wide range of medical information.

While the proposed encryption scheme demonstrates strong resistance to traditional cryptographic attacks, it is also designed to be resilient against modern threats such as machine learning-based cryptanalysis such as predicting encryption keys and breaking encryption schemes. However, the high randomness introduced by the chaotic maps and the complexity of fractal-based key generation in our approach makes it difficult for machine learning algorithm to successfully predict the key or decrypt image.

Conclusion

In this work we have introduced a novel approach for enhancing image encryption through the integration of chaotic systems, fractal patterns and advanced cryptographic techniques. The proposed method leverages the May map to generate a robust S-box which is then employed to substitute the pixel values in red, green and blue channels of an image. This substitution introduces significant non-linearity and complexity into the encryption process thereby strengthening the security of the image data. This hybrid approach ensures a high level of security by expediting the chaotic nature of the maps and complexity of rectal patterns. The strength of our encryption scheme lies in its ability to combine chaotic and fractal-based approaches which significantly enhances both the randomness and the security of the encryption process.

Future work

The utilization of chaotic maps, and fractal patterns for image encryption is computationally efficient, making it suitable for IoT devices with limited processing capabilities. Studies have demonstrated the feasibility of real-time medical image encryption using improved sequence from chaotic maps, highlighting its applicability in resource-contrainted settings[33]. However, challenges such as scalability and interoperability must be addressed to ensure robust application. To tackle these issues we plan to develop lightweight encryption modules optimized for low-power devices, ensuring efficient performance without compromising security. The future work will focus on scalable blockchain architectures and comprehensive testing within real-world IoT medical environment to validate the efficiency of proposed scheme. Moreover, future work will explore the application of this technique to different types of data and access their performance in various cryptographic context aiming to further refine and optimize the encryption process.

Data availability

The data used to support the findings of this study are included within the article.

Received: 29 October 2024; Accepted: 6 February 2025
Published online: 05 March 2025

References

1. Akpakwu, G. A., Silva, B. J., Hancke, G. P. & Abu-Mahfouz, A. M. A survey on 5G networks for the Internet of Things: Communication technologies and challenges. *IEEE Access* **6**, 3619–3647 (2017).
2. Purohit, M. K. Application of cryptography using artificial intelligence.
3. Khan, P. W. & Byun, Y. A blockchain-based secure image encryption scheme for the industrial Internet of Things. *Entropy* **22**(2), 175 (2020).
4. Sharma, N., Shamkuwar, M., & Singh, I. (2019). The history, present and future with IoT. *Internet of Things and Big Data Analytics for Smart Generation*, 27–51.
5. Chataut, R., Phoummalayvane, A. & Akl, R. Unleashing the power of IoT: A comprehensive review of IoT applications and future prospects in healthcare, agriculture, smart homes, smart cities, and industry 4.0. *Sensors* **23**(16), 7194 (2023).
6. Inam, S., Kanwal, S., Firdous, R. & Hajjej, F. Blockchain based medical image encryption using Arnold's cat map in a cloud environment. *Sci. Rep.* **14**(1), 5678 (2024).
7. Ahram, T., Sargolzaei, A., Sargolzaei, S., Daniels, J., & Amaba, B. (2017). Blockchain technology innovations. In *2017 IEEE Technology & Engineering Management Conference (TEMSCON)* (pp. 137–141). IEEE.
8. Laroiya, C., Saxena, D., & Komalavalli, C. (2020). Applications of blockchain technology. In *Handbook of Research on Blockchain Technology* (pp. 213–243). Academic Press.
9. Li, D., Luo, Z. & Cao, B. Blockchain-based federated learning methodologies in smart environments. *Clust. Comput.* **25**(4), 2585–2599 (2022).
10. Alexan, W. et al. AntEater: When Arnold's cat meets langton's ant to encrypt images. *IEEE Access* **11**, 106249–106276. https://doi.org/10.1109/ACCESS.2023.3319335 (2023).
11. Alexan, W. et al. Stegocrypt: A robust tri-stage spatial steganography algorithm using TLM encryption and DNA coding for securing digital images. *IET Image Proc.* https://doi.org/10.1049/ipr2.13242 (2024).
12. Gabr, M. et al. R3—Rescale, rotate, and randomize: A novel image cryptosystem utilizing chaotic and hyper-chaotic systems. *IEEE Access* **11**, 119284–119312. https://doi.org/10.1109/ACCESS.2023.3326848 (2023).
13. Alexan, W., El-Damak, D. & Gabr, M. Image encryption based on fourier-DNA coding for hyperchaotic chen system, chen-based binary quantization S-box, and variable-base modulo operation. *IEEE Access* **12**, 21092–21113. https://doi.org/10.1109/ACCESS.2024.3363018 (2024).
14. Gabr, M. et al. Application of DNA coding, the lorenz differential equations and a variation of the logistic map in a multi-stage cryptosystem. *Symmetry* **14**(12), 2559. https://doi.org/10.3390/sym14122559 (2022).

15. Alexan, W., Gabr, M., Mamdouh, E., Elias, R. & Aboshousha, A. Color image cryptosystem based on sine chaotic map, 4D chen hyperchaotic map of fractional-order and hybrid DNA coding. *IEEE Access* **11**, 54928–54956. https://doi.org/10.1109/ACCESS.2023.3282160 (2023).
16. El-Damak, D. et al. Fibonacci Q-matrix, hyperchaos, and galois field (2^8) for augmented medical image encryption. *IEEE Access* **12**, 102718–102744. https://doi.org/10.1109/ACCESS.2024.3433499 (2024).
17. Feixiang, Z., Mingzhe, L., Kun, W. & Hong, Z. Color image encryption via Hénon-zigzag map and chaotic restricted Boltzmann machine over Blockchain. *Opt. Laser Technol.* **135**, 106610 (2021).
18. Bhaskaran, R., Karuppathal, R., Karthick, M., Vijayalakshmi, J., Kadry, S., & Nam, Y. (2022). Blockchain enabled optimal lightweight cryptography based image encryption technique for IIoT. *Intell. Autom. Soft Comput*. **33**(3).
19. Hazzazi, M. M., Baowidan, S. A., Yousaf, A. & Adeel, M. An innovative algorithm based on chaotic maps amalgamated with bit-level permutations for robust S-box construction and its application in medical image privacy. *Symmetry* **16**(8), 1070 (2024).
20. Kanwal, S., Inam, S., Quddus, S. & Hajjej, F. Research on color image encryption approach based on chaotic duffing map. *Physica Scripta* **98**(12), 125252 (2023).
21. Zhikharev, L. A. (2021). A Sierpiński triangle geometric algorithm for generating stronger structures. *J. Phys.: Conf. Ser.* (vol. 1901, no. 1, p. 012066). IOP Publishing.
22. Ali, A. et al. A fractal-based authentication technique using sierpinski triangles in smart devices. *Sensors* **19**(3), 678 (2019).
23. Kmader. (n.d.). SIIM Medical Images. Kaggle. Retrieved from https://www.kaggle.com/datasets/kmader/siim-medical-images
24. Midouazerty. (2022). Breast cancer images classification. Kaggle. https://www.kaggle.com/code/midouazerty/breast-cancer-images-classification/input
25. Abdelsamie, F. E. A hybrid approach for medical image fusion based on wavelet transform and principal component analysis. *Menoufia J. Electron. Eng. Res.* **27**(2), 59–70 (2018).
26. Chen, Y.-P., Fan, T.-Y. & Chao, H.-C. WMNet: A lossless watermarking technique using deep learning for medical image authentication. *Electronics* **10**, 932 (2021).
27. Natiq, H., Al-Saidi, N., Said, M. & Kilicman, A. A new hyperchaotic map and its application for image encryption. *Eur. Phys. J. Plus* **133**, 6 (2018).
28. Ahmad, J. & Hwang, S. O. A secure image encryption scheme based on chaotic maps and affine transformation. *Multimed. Tools Appl.* **75**, 13951–13976 (2016).
29. Zhou, S. et al. Encryption method based on a new secret key algorithm for color images. *AEU Int. J. Electron. Commun.* **70**, 1–7 (2016).
30. Belazi, A., El-Latif, A. A. A. & Belghith, S. A novel image encryption scheme based on substitution-permutation network and chaos. *Signal Process.* **128**, 155–170 (2016).
31. Patro, K. A. & Acharya, B. (2021). An efficient dual-layer cross-coupled chaotic map security-based multi-image encryption system. *Nonlinear Dyn*. 104. https://doi.org/10.1007/s11071-021-06409-z.
32. Xudong Liu, Xiaojun Tong, Zhu Wang, Miao Zhang, Yunhua Fan. A novel devaney chaotic map with uniform trajectory for color image encryption. Appl. Math. Model. **120**, 2023, pp. 153–174, ISSN 0307-904X.
33. Trujillo-Toledo, D. A., López-Bonilla, O. R., García-Guerrero, E. E., Esqueda-Elizondo, J. J., Cárdenas-Valdez, J. R., Tamayo-Pérez, U. J., Aguirre-Castro, O. A., Inzunza-González, E. Real-time medical image encryption for H-IoT applications using improved sequences from chaotic maps. *Integration*, vol. 90, 2023, pp. 131–145, ISSN 0167-9260.
34. Inam, S. et al. Blockchain based medical image encryption using Arnold's cat map in a cloud environment. *Sci Rep* **14**, 5678 (2024).

Acknowledgements

Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R104), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Author contributions

1. Conceptualization, Methodology, Visualization, Validation, supervision: Saba Inam 2. Conceptualization, Methodology, Visualization, Validation: Shamsa Kanwal 3. Writing Original Draft and Experimental Results: Mamoona Batool 4.Validation, Review and Editing: Shaha Al Otaibi 5. Additional Experimental results, Review and editing: Mona M. Jamjoom.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to S.I.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.