Journal of Ambient Intelligence and Humanized Computing (2023) 14:1731–1752
https://doi.org/10.1007/s12652-021-03410-7

ORIGINAL RESEARCH

A medical image cryptosystem using bit-level diffusion with DNA coding

Pooja Mishra[1] · Chiranjeev Bhaya[1] · Arup Kumar Pal[1] · Abhay Kumar Singh[2]

Received: 6 November 2020 / Accepted: 6 July 2021 / Published online: 26 August 2021

Abstract

This paper presents a secured encryption algorithm for medical images using the concept of DNA cryptography. It proposes a novel technique of masking the images before encryption, which is a keyless process but aids in increasing the original image's randomness. Confusion and diffusion are then performed on the masked image, due to which the cipher image has no perceptual or statistical information. The proposed algorithm uses generalized Arnold's Cat Map for the confusion. In addition, a novel diffusion process has been introduced, which operates on both pixel level and DNA-plane level. It incorporates all possible DNA encoding, decoding, and XOR rules, selected pseudo-randomly based on chaotic 2D-Logistic Sine Coupling Map values.Thus, it makes the cipher image more robust against brute force and statistical attacks and almost impossible for the intruder to obtain the original image without knowing the correct key. However, the original image can be deciphered using the valid key without any data loss, which is very important for medical images. A single round of masking, confusion, and diffusion steps are enough to obtain the cipher image. The proposed algorithm has been tested over many medical and natural images and on homogeneous and textured patterns. The cipher images obtained have a low inter-pixel correlation coefficient of around 0 and high entropy of close to 8 bits/symbol. Moreover, the analysis of the proposed method on other parameters like key-space, key-sensitivity, cipher statistics, and differential, occlusion, and noise attacks also gives satisfactory results required for a secure image cryptosystem.

Keywords 2D-Logistic Sine Coupling Map · Confusion · Diffusion · DNA encryption · Masking · Medical image cryptosystem

1 Introduction

Medical images play an essential role in health care sectors to provide an accurate diagnosis of disease, therapy plans, and treatment follow-ups. Different imaging technologies such as Ultrasound (US), Magnetic Resonance Imaging (MRI), Computed Tomography (CT) (Parvees et al. 2016), and Microscopy (Połap 2020) are used to generate medical images. Modern techniques like telemedicine, teleradiology, and telesurgery that provide remote healthcare facilities for patients require secured medical data transmission (Dagadu et al. 2019). For example, in teleradiology, medical images provide the patient's complete medical information in digital image form. Therefore a tremendous amount of image data and medical information is produced by public and private health care organizations (Wang et al. 2018a).

A large number of medical images are also used to study and diagnose various diseases with the help of advanced computational techniques like feature extraction and image processing (Połap and Srivastava 2020) using deep learning techniques (Haskins et al. 2020; Połap 2019). The medical data were previously stored in local storage systems, which are now replaced by the Picture Archiving and Communication System (PACS) (Li et al. 2020). Moreover, these images

✉ Pooja Mishra
poojatrip2002@gmail.com

Chiranjeev Bhaya
cbhaya@gmail.com

Arup Kumar Pal
arupkrpal@gmail.com

Abhay Kumar Singh
itbhu81@gmail.com

[1] Department of Computer Science and Engineering, Indian Institute of Technology (Indian School of Mines), Dhanbad, India

[2] Department of Mathematics and Computing, Indian Institute of Technology (Indian School of Mines), Dhanbad, India

are also used for insurance claims and medical researches. Therefore, it is essential to ensure the confidentiality and integrity of the medical images. (Khare and Srivastava 2020).

Since images have larger data capacity and more data redundancy than text data, traditional symmetric-key algorithms such as AES and DES (Rijmen and Daemen 2001; Smid and Branstad 1988) cannot be used to encrypt them. Moreover, since adjacent pixels in images are highly correlated, cryptosystems should be designed to reduce perceptual information in the cipher image. Therefore, image encryption algorithms incorporate different techniques such as the use of chaotic systems (Fridrich 1997; Qiao et al. 2020), DNA sequence (Belazi et al. 2019), and cellular automata (Wang et al. 2018b). Most of the cryptosystems follow confusion-diffusion architecture. In the confusion process, pixels are shuffled from their positions. Thus, the visual perception of the original image gets changed. Chaotic maps such as Arnold Cat map, logistic map, and Henon map are some techniques used in the confusion process (Wu et al. 2017). In the diffusion process, the intensity values of the pixels are changed (Akkasaligar and Biradar 2020). Mostly chaotic maps and some operations like addition, subtraction, and XOR are used to diffuse the original image to remove its statistical information. The sequence of confusion and diffusion are iterated over themselves several times.

DNA computing is a new paradigm to encrypt data due to its high storage capacity, low power consumption, and vast parallelism. A DNA molecule is composed of four nucleotides, namely Adenine (A), Guanine (G), Thymine (T) and Cytosine (C). The sequence of these nucleotides in a DNA string stores the information. In most DNA-based cryptosystems, the image is converted into a string of DNA nucleotides and algebraic operations such as addition, subtraction, and XOR. Permutations are also carried out on these strings. These nucleotides are then converted back to an image to form a cipher image. The cipher image can be decrypted back to the original image by following the corresponding decoding algorithm using the correct key. In the paper of Dua et al. (2019), the color image was encrypted using the DNA encoding and decoding rules, differential optimization technique, and interwining logistic map. The method followed the confusion-diffusion architecture. In the paper of El-Khamy and Mohamed (2021), the authors proposed an encryption algorithm to secure the gray images with the use of DNA encoding and decoding rules, and one-dimensional logistic map, and Chen's hyperchaotic map. The original image was first decomposed into an eight-bit binary image, and the logistic map sequences were used to scramble the binary image. The confused image was encoded into DNA sequences, and the DNA image was split into four A, T, G, C sub-images. These sub-images were XORed with the application of Chen's hyperchaotic map sequences, and again these four diffused sub-images were fused with the Discrete Wavelet Transform (DWT).

Medical images have fine and informative features as compared to natural images. Therefore, any loss of information should be avoided. Since the nature of the medical image is different, the encryption methods must be changed (Jeevitha and Prabha 2020; Tsafack et al. 2020). In the paper of Jeevitha and Prabha (2020), the image was decomposed into n numbers of Discrete Wavelet Transform (DWT) planes, and the Edge Map was used to generate the same number of edge sequences. These DWT-planes and edge sequences were XORed, and again the diffused DWT planes were permutated, and the cipher image was constructed. In the whole process, the double-Sine map was applied to generate the chaotic sequences to provide randomness. Tsafack et al. (2020) presented an encryption method with the application of Hamming distance, Mandelbrot set, and the Logistic-Sine Coupling map. Only diffusion processes were applied to obtain the cipher image. The algorithms of (Akkasaligar and Biradar 2016, 2018, 2020; Belazi et al. 2019; Dagadu et al. 2019; Parameshachari et al. 2017; Stalin et al. 2019) have been designed using various DNA operations. Akkasaligar and Biradar (2018), extracted DNA sequences of *Canis cupus* of variable length and applied SHA-256 to generate the key. Using Chen's hyperchaotic map, a DNA XOR operation was performed for the diffusion of the plain image. Akkasaligar and Biradar (2016), in their paper, considered odd and even pixels respectively to separate the images into two matrices and encoded the image using two different DNA encoding rules. Lorenz and Chen's chaotic maps were used for the confusion and diffusion process. A sorted sequence from the maps was used in the encryption process. It is though a very complex operation. DNA ADD was used in encryption, and DNA SUB was used in the decryption process. Akkasaligar and Biradar (2016), Akkasaligar and Biradar (2018) used only one rule for DNA XOR, DNA ADD, and DNA SUB. Belazi et al. (2019) proposed an encryption technique where the key was generated from the plain image using SHA-256. DNA encoding, decoding, and complementary operations were performed using a logistic-Chebyshev map, whereas a bitwise XOR operation was performed using a sine-Chebyshev map. However, for generating a secured cipher key, two rounds of encryption had to be performed. Dagadu et al. (2019) proposed a color medical image encryption technique using Bernoulli-shift map and zigzag map. MD5 was used for generating the initial conditions of the map. DNA XOR operation was used for diffusion, and a logistic map was used to select DNA encoding and decoding rules. One of the disadvantages of using hash functions for generating key-stream is high computational operations, mainly when images of high dimensions are used. In the paper of Akkasaligar and Biradar (2020), the medical image was first partitioned into two 8-bit matrices (M_1, M_2) having selected pixels of the image and non selected pixels of the image, respectively. Then the matrices M_1 and M_2 were encoded into DNA sequences using randomly chosen

DNA encoding rules, and then the dual hyperchaotic map was applied to permutate the matrix M_1. The shuffled matrices M_1' and M_2 were DNA XORed, and the resultant diffused DNA sequence matrix was converted into a binary matrix with the DNA decoding rules to obtain the cipher image. Chen et al. (2018) introduced a self-adaptive permutation-diffusion technique and applied DNA encoding-decoding rules for the medical image encryption. To ensure the randomness of the cryptosystem Hyperchaotic Lorenz system was used, and the algorithm was iterated n-rounds for the security of the images.

Motivated by the above discussions, a new cryptosystem has been designed to remove some imperfections of the existing algorithms. Most of the algorithms follow the traditional confusion-diffusion architecture iterating over itself for several rounds. Only a few DNA encoding or decoding rules have been used in static form, which is an underutilization of DNA properties. Moreover, all cryptosystems for natural images may not be equally efficient for medical images as they have higher pixel correlation, lower entropy than the former, and need to be decrypted without any loss of information. The proposed cryptosystem, therefore, has the following distinctiveness.

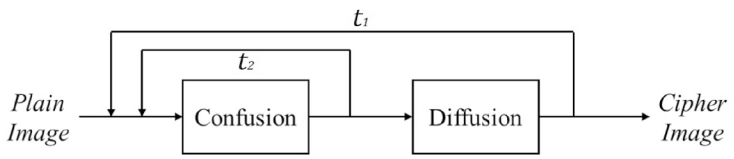
- Firstly, the original image is masked with a constant vector iterating over itself. Although this method is keyless, it increases the randomness of the image and enhances the cryptosystem's efficiency.
- ACM has been used to confuse the masked image so that the visual perception is lost. Since ACM has a periodicity, the number of rounds has been decided so that the resultant image is highly confused. Moreover, generalized ACM has been used, the values of which are derived from the input key.
- 2D Logistic-Sine-Coupling-Map (2D-LSCM) has been used for key scheduling. The map produces a pseudo-random sequence that can be expanded according to the size of the image. Therefore, a key of 256-bits is enough for encryption of large-size images as well.
- Bit-level permutation is performed on every pixel so that information in every significant bit of the image is equally distributed.
- All the eight DNA encoding, decoding, and XOR rules are used in the diffusion algorithm, the choice of which is determined by the input map. Therefore, the map can be derived only from the correct key, making the algorithm even more secure and robust against statistical and differential attacks. It also makes the key sensitivity relatively high.
- The entire process of masking, confusion, and diffusion is carried out only once, with the confusion process having iterations contrary to the multiple iterations in the conventional approach, therefore making the algorithm computationally cost-efficient as illustrated in the following Fig.1.

The framework of the remaining part of the paper is as follows: Sect. 2 describes some preliminaries on ACM, DNA encoding, decoding, and XOR rules which have been used in the encryption algorithm. Section 3 gives a brief description of the proposed cryptosystem, which is based on both pixel-level shuffling and bit-level diffusion by using DNA-planes and 2D-Logistic Sine Coupling Map (2D-LSCM). The cryptosystem was tested on several images, and security analysis was performed on them, which has been illustrated in Sect. 4. Comparisons with some existing algorithms have been shown in Sect. 5. Finally, conclusion has been drawn in Sect. 6.

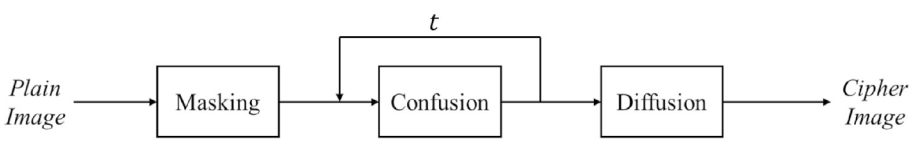
2 Preliminaries

2.1 Arnold's Cat Map

Arnold's Cat Map (ACM) (Arnol'd and Avez 1968) is a chaotic map that is used for shuffling an image. For a matrix of size $N \times N$, the generalized ACM is represented as



(a) Conventional image encryption approach involving a number of rounds of confusion followed by diffusion iterating over themselves.



(b) Proposed algorithm involving a single round of masking followed by iterative confusion and a single round of diffusion, thus reducing time complexity.

Fig. 1 Conventional approach of image encryption *vs* proposed approach

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & \alpha \\ \beta & (1+\alpha\beta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \pmod{N} \quad (1)$$

where $x, y, x', y' \in [1, N]$. (x, y) and (x', y') represent the position vectors of the original image and the shuffled image respectively. α and β are two positive integers.

ACM is invertible. Since its determinant is one, the inverse always results in integral values. The inverse of ACM is given by

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} (1+\alpha\beta) & -\alpha \\ -\beta & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} \pmod{N} \quad (2)$$

Here, α and β may be considered as secured parameters, and the values can be obtained from the secret key.

2.2 DNA encoding and decoding rules

DNA encoding is the process of representing data (here image) as a sequence of DNA nucleotides, namely *A*, *C*, *T*, and *G*. Encoding is done based on Watson-Crick Complement (WCC) rules, where (*A*, *T*) and (*G*, *C*) are complementary pairs (Kari and Mahalingam 2007). Out of 24 possible combinations for DNA encoding rules, only eight of them will follow the WCC rules. Table 1 shows the eight possible encoding rules.

The DNA decoding rules are just the inverse of the encoding rules. Therefore, the DNA decoding rules corresponding to the eight encoding rules are summarized in Table 2.

2.3 DNA XOR rules

Corresponding to the eight DNA encoding rules, DNA XOR rules can be formulated. Table 3 shows the eight DNA XOR rules. The XOR rules are invertible. For example, if d_1, d_2 and d_3 are three DNA sequences and $d_1 \oplus d_2 = d_3$, then $d_1 = d_3 \oplus d_2$ and $d_2 = d_3 \oplus d_1$. These rules are used in the diffusion algorithm. Owing to the properties of XOR, the same algorithm can be used for encryption and decryption as well.

Table 1 DNA encoding rules

Quad value	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8
0	*A*	*A*	*C*	*G*	*C*	*G*	*T*	*T*
1	*C*	*G*	*A*	*A*	*T*	*T*	*C*	*G*
2	*G*	*C*	*T*	*T*	*A*	*A*	*G*	*C*
3	*T*	*T*	*G*	*C*	*G*	*C*	*A*	*A*

Table 2 DNA decoding rules

Nucleotide	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8
A	0	0	1	1	2	2	3	3
C	1	2	0	3	0	3	1	2
G	2	1	3	0	3	0	2	1
T	3	3	2	2	1	1	0	0

Table 3 DNA XOR rules

	Rule 1				Rule 2				Rule 3				Rule 4			
$\oplus$	*A*	*C*	*G*	*T*	*A*	*C*	*G*	*T*	*A*	*C*	*G*	*T*	*A*	*C*	*G*	*T*
A	*A*	*C*	*G*	*T*	*A*	*C*	*G*	*T*	*C*	*A*	*T*	*G*	*G*	*T*	*A*	*C*
C	*C*	*A*	*T*	*G*	*C*	*A*	*T*	*G*	*A*	*C*	*G*	*T*	*T*	*G*	*C*	*A*
G	*G*	*T*	*A*	*C*	*G*	*T*	*A*	*C*	*T*	*G*	*C*	*A*	*A*	*C*	*G*	*T*
T	*T*	*G*	*C*	*A*	*T*	*G*	*C*	*A*	*G*	*T*	*A*	*C*	*C*	*A*	*T*	*G*
	Rule 5				Rule 6				Rule 7				Rule 8			
$\oplus$	*A*	*C*	*G*	*T*	*A*	*C*	*G*	*T*	*A*	*C*	*G*	*T*	*A*	*C*	*G*	*T*
A	*C*	*A*	*T*	*G*	*G*	*T*	*A*	*C*	*T*	*G*	*C*	*A*	*T*	*G*	*C*	*A*
C	*A*	*C*	*G*	*T*	*T*	*G*	*C*	*A*	*G*	*T*	*A*	*C*	*G*	*T*	*A*	*C*
G	*T*	*G*	*C*	*A*	*A*	*C*	*G*	*T*	*C*	*A*	*T*	*G*	*C*	*A*	*T*	*G*
T	*G*	*T*	*A*	*C*	*C*	*A*	*T*	*G*	*A*	*C*	*G*	*T*	*A*	*C*	*G*	*T*

2.4 DNA planes

An image is composed of a number of pixels, and every pixel is made of some bits. On encoding, every two bits of a pixel with some DNA encoding rule as defined in Table 1, a string of nucleotides will be formed having a length half of the number of bits in a pixel. For example, a typical grayscale image composed of eight bits per pixel will contain four nucleotides per pixel on DNA encoding.

A DNA plane is a set of nucleotides corresponding to a given nucleotide position in each DNA sequence representing every image's pixel. For a grayscale image of eight-bit depth, four DNA planes can be obtained. Figure 2 shows the resultant images when decomposed into DNA planes.

3 Proposed algorithm

The proposed algorithm for medical image encryption uses 2D-LSCM (Hua et al. 2018) to generate chaotic pseudo-random sequences required for various operations. The initial parameters of the map are given by the key sequence, which is 256-bit long. Since the same key is required for both encryption as well as decryption, the key is symmetric. The encryption algorithm consists of masking, confusion, and DNA-plane-based diffusion processes, and the decryption process is just the reverse of encryption. This section briefly discusses the various steps involved in the proposed algorithm.

3.1 Key scheduling and chaotic maps

The main confusion-diffusion of the cryptosystem is carried out using 2D-LSCM as formulated and described by Hua et al. (2018). Five such 2D-LSCMs, $(M^{(0)}, \ldots, M^{(4)})$ have been used, the formation of which requires the symmetric key.

The 2D-LSCM is defined by Hua et al. (2018) as

$$\begin{cases} x_{i+1} = \sin(\pi(4\theta x_i(1-x_i) + (1-\theta)\sin(\pi y_i))); \\ y_{i+1} = \sin(\pi(4\theta y_i(1-y_i) + (1-\theta)\sin(\pi x_{i+1}))); \end{cases} \tag{3}$$

where θ is the control parameter, $\theta = [0, 1]$.

The secret key K, of length 256 bits, is defined as

$$K = \{x_0, y_0, \theta, a_1, a_2, a_3, a_4\} \tag{4}$$

The parameters are defined as follows.

- (x_0, y_0) are 52 bits each, determining the initial values of $M^{(0)}$. These values are converted to IEEE-754 floating-point standard (Stevenson 1981). Let $b_1 b_2 b_3 ... b_{52}$ be the binary representation; its equivalent floating-point value is calculated as

$$\sum_{i=1}^{52} b_i \cdot 2^{-i} \tag{5}$$

- The initial values of $M^{(1)} ... M^{(4)}$ are calculated from the final values of the previous maps, i.e., $(x_0^{(1)}, y_0^{(1)}) = (x_f^{(0)}, y_f^{(0)})$, $(x_0^{(2)}, y_0^{(2)}) = (x_f^{(1)}, y_f^{(1)})$, $(x_0^{(3)}, y_0^{(3)}) = (x_f^{(2)}, y_f^{(2)})$, and $(x_0^{(4)}, y_0^{(4)}) = (x_f^{(3)}, y_f^{(3)})$.
- $(x_f^{(4)}, y_f^{(4)})$ is used as the seed value for ACM. These seed values are fed into some random number generator to obtain the integral values of α and β to be used in Eq. 1 for ACM.
- θ is a 52 bit value, converted to floating-point value as shown in Eq. 5. It provides the initial control parameter for $M^{(0)}$.
- $a_1, \ldots, a_4$ are 25 bits each, used to determine the control parameters $\theta^{(1)}, \ldots, \theta^{(4)}$ of maps $M^{(1)}, \ldots, M^{(4)}$ respectively as

(a) Original Image **(b)** DNA Plane 4 **(c)** DNA Plane 3 **(d)** DNA Plane 2 **(e)** DNA Plane 1

Fig. 2 Decomposition of an image into four DNA planes

- $\theta^{(1)} = \texttt{FRACTION}(\theta \times a_1)$
- $\theta^{(2)} = \texttt{FRACTION}(\theta^{(1)} \times a_2)$
- $\theta^{(3)} = \texttt{FRACTION}(\theta^{(2)} \times a_3)$
- $\theta^{(4)} = \texttt{FRACTION}(\theta^{(3)} \times a_4)$

where $\texttt{FRACTION}(x)$ returns the fractional part of x.

3.2 Encryption algorithm

The encryption algorithm converts the plain image into a cipher image. Once the chaotic sequences are obtained from the key scheduling algorithm, the values obtained from them are used in the various encryption steps. Figure 3 summarizes the various steps of encryption that begin with masking and is followed by confusion and DNA-plane-based diffusion steps, which removes all the statistical and visual information from the original image and generates a random-noise-like cipher. The following subsections discuss the various steps involved in the encryption algorithm.

3.2.1 Masking

Medical images generally contain lower entropy compared to natural images. Masking is applied over the original image to increase the entropy. In this step, the original image is XORed with a constant vector from 0 to 255, iterating over itself to generate the masked image, which is then used in the subsequent encryption steps. The masking algorithm is described in Algorithm 1.

Algorithm 1: Masking

Input : Original Image I of size $m \times n$
Output: Masked Image M of size $m \times n$

count $\leftarrow 0$
for $x \leftarrow 1$ **to** m **do**
  for $y \leftarrow 1$ **to** n **do**
    $M[x][y] \leftarrow I[x][y] \oplus$ count
    count $\leftarrow$ (count $+1$) mod 256
return M

3.2.2 Confusion using ACM

Arnold's Cat Map is used for confusion, as is described in Sect. 2.1. The seed value $seed = (x_f^{(4)}, y_f^{(4)})$ is obtained from the 2D-LSCM as described in Sect. 3.1 which is then fed to some random integer generator $\texttt{RAND}()$, giving the values of α and β for ACM. The steps are repeated several times to obtain a properly shuffled image, which eventually generates an image that loses most of the visual information. The confusion algorithm is described in Algorithm 2.

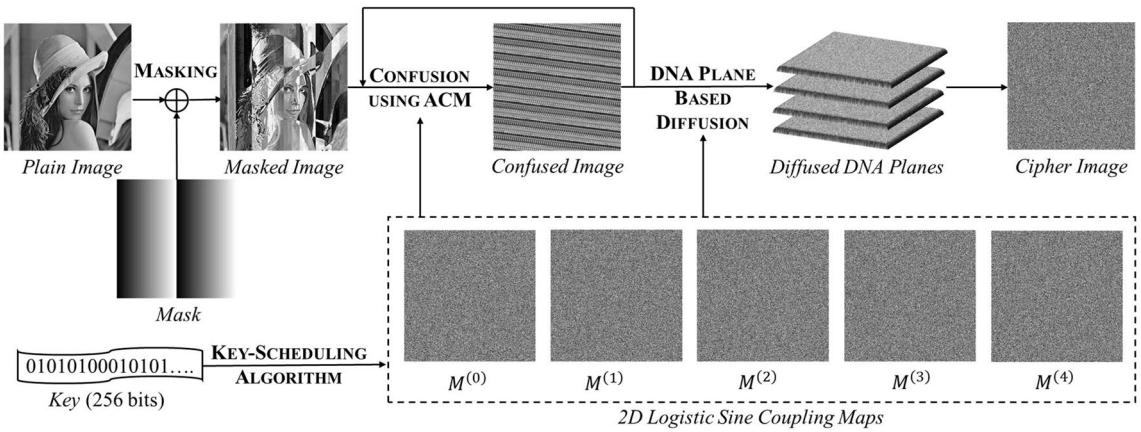


Fig. 3 Encryption algorithm

Algorithm 2: Confusion using ACM

Input : Masked Image M of size $m \times n$, seed value *seed*
Output: Shuffled Image M' of size $m \times n$
Let p be the largest prime-number $\leq \lfloor m/2 \rfloor$
$\alpha, \beta = \texttt{RAND}(seed)$ `// RAND() is some random integer generator function`
for $x \leftarrow 1$ **to** p **do**
  $M' \leftarrow \texttt{ACM}(M, \alpha, \beta)$ `// using Equation 1`
  $M \leftarrow M'$
return M'

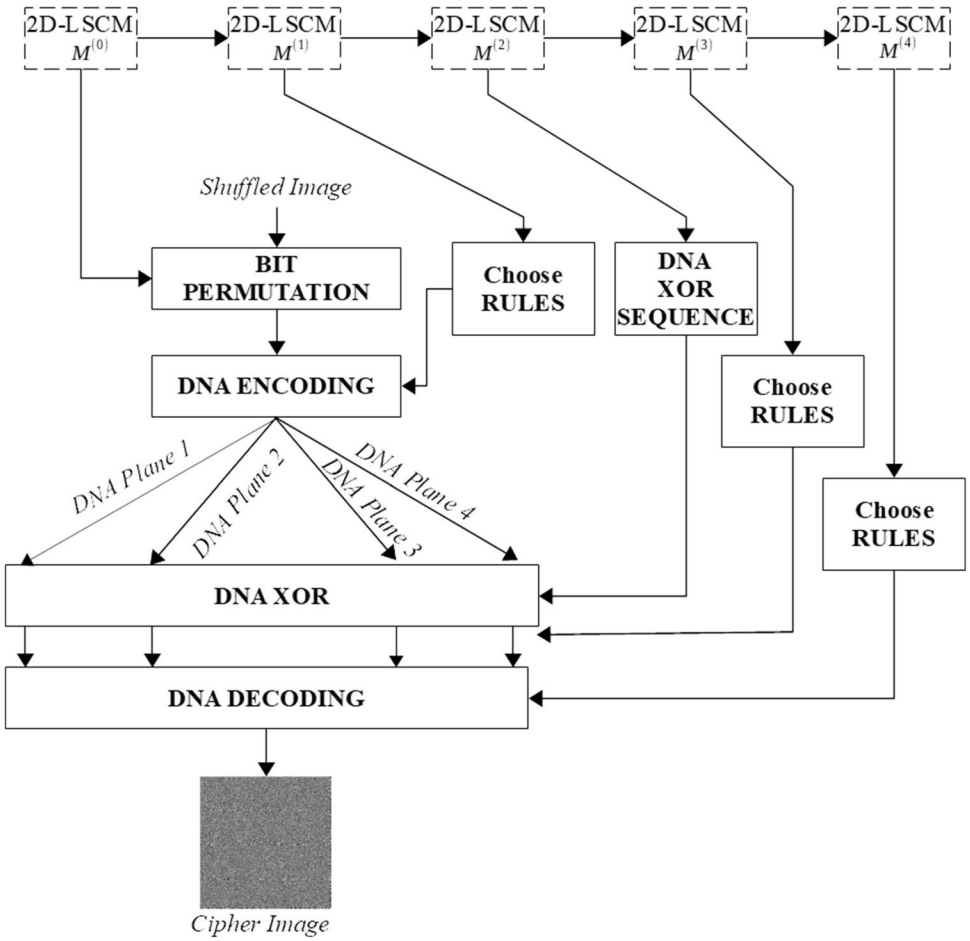


Fig. 4 Flowchart of the diffusion architecture

3.2.3 DNA-plane based diffusion

The diffusion process is the heart of the proposed cryptosystem. The shuffled image obtained from the confusion state loses all its visual information, yet some statistical data is preserved, making it vulnerable to attacks. The diffusion process removes all the statistical information from the image so that there is no similarity between the input and the output image. The diffusion step first uses bit-level scrambling followed by DNA encoding, which divides the image into four DNA planes. Each plane is then XORed with a chaotic sequence obtained from one of the maps. These XORed sequences are combined back and using DNA decoding, and the corresponding cipher image is obtained. The chaotic map sequences are used to determine the XOR rules. The diffusion process is illustrated using the flowchart in Fig. 4, and the methods are discussed as follows (Fig. 5).

– *Bit permutation :* This step takes a shuffled image obtained from Algorithm 2 as an input along with the

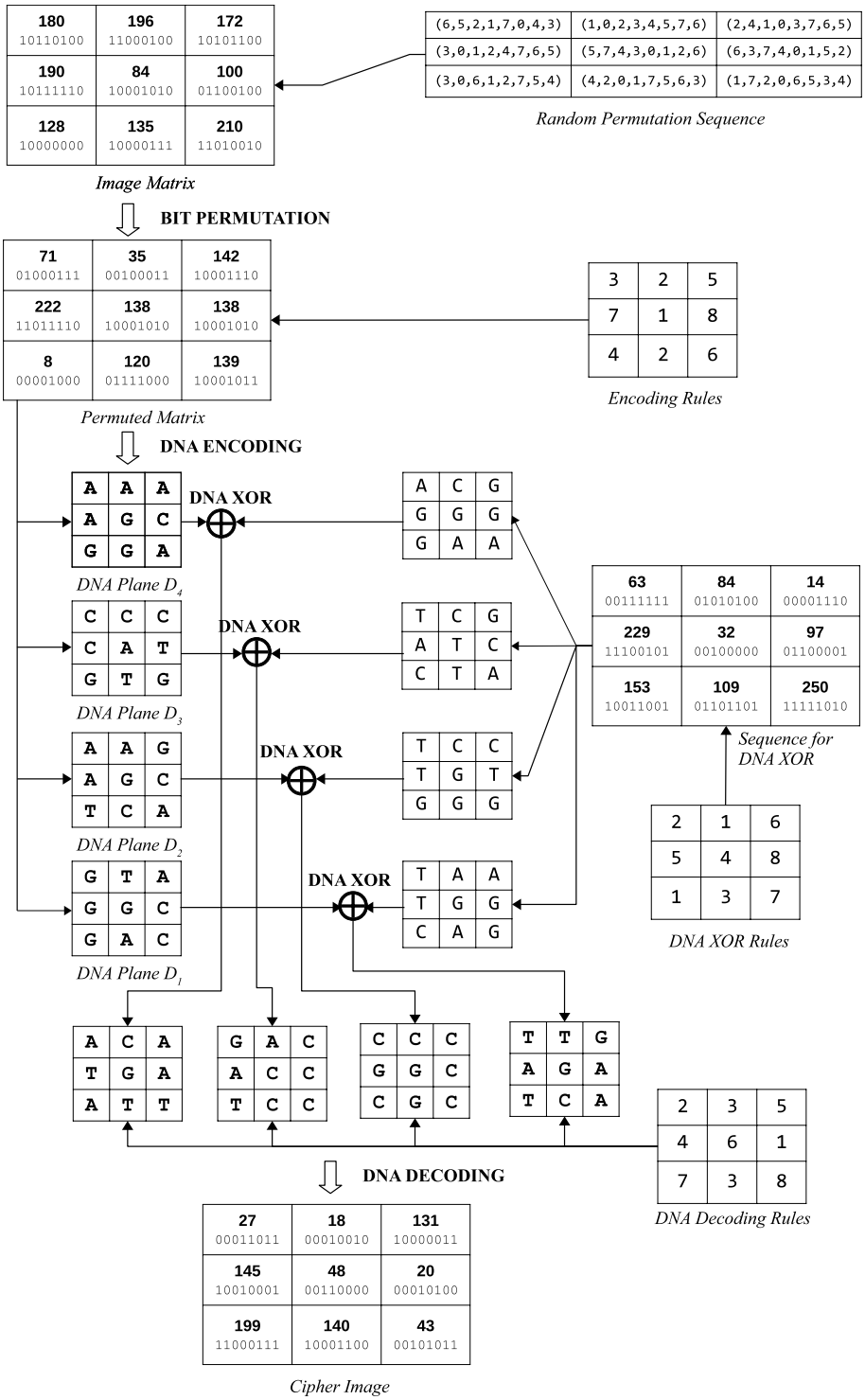


Fig. 5 Illustration of the DNA-plane based diffusion process

chaotic sequence of the 2D-LSCM $M^{(0)}$. The map is expanded to the image's size, and the pseudo-random sequences corresponding to each of the pixels of the shuffled image are fed to a random permutation function named RANDOM_PERMUTE(). This function generates a permutation of integers from 0 to 7 corresponding to which the bits of every pixel of the image are shuffled. The algorithm for bit permutation is presented in Algorithm 3.

Figure 6 shows an example of bit-permutation. The integer value of a pixel is converted into 8-bit binary sequence b. The permutation value at the i^{th} position, say

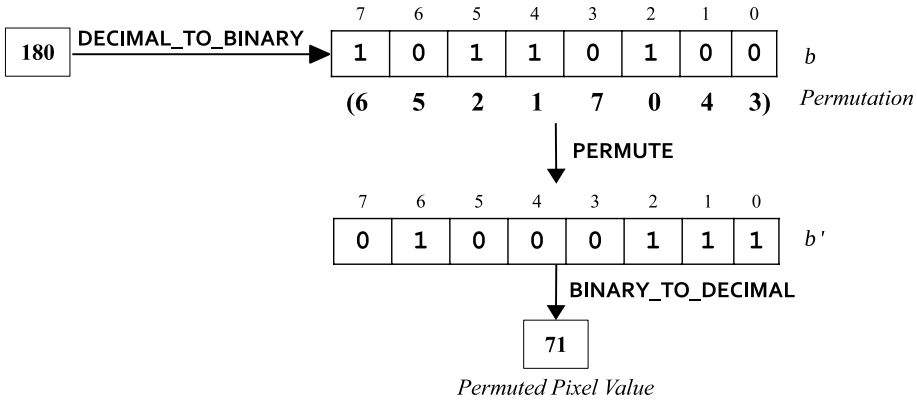


Fig. 6 An example of bit permutation

$p[i]$, maps the value to b' using the equation $b'[p[i]] = b[i]$ to form the permuted binary sequence b'. From b' and p, the original sequence can be obtained back using the reverse permutation as $b[i] = b'[p[i]]$. The binary string b' is converted to decimal to obtain the permuted pixel value.

- *Choose rules:* This process generates the rule sequence for DNA encoding, decoding, and XOR. The chaotic map is expanded to the size of the image. Since the map values are in the range of [0, 1], they are converted to a value from 1 to 8 by multiplying with eight and adding 1 to the resultant integer part. The algorithm for choosing rules is described in Algorithm 4.
- *DNA encoding:* The DNA encoding process converts the permuted image obtained as a result of Algorithm 3 to four DNA planes according to the rules chosen from the rule sequence obtained from Algorithm 4 using 2D-LSCM $M^{(1)}$. The encoding process's result is four DNA planes containing nucleotides of the same dimension as the image. The DNA encoding algorithm is presented in Algorithm 5.
- *DNA XOR sequence :* For the DNA XOR, a pseudo-random sequence of the integer is required of size as large as that of the image. For this, the values from 2D-LSCM $M^{(2)}$ is expanded to the size $\left\lceil \frac{m \times n}{256} \right\rceil$. These values are one by one fed as the seed value of some random permutation function RANDOM_PERMUTE() to generate a pseudo-random sequence from 0 to 255. These sequences are used for DNA XOR. The algorithm for obtaining DNA XOR sequence is presented in Algorithm 6.
- *DNA XOR:* In this process, the DNA planes are divided into 1×4 non-overlapping blocks. The DNA XOR sequence obtained from Algorithm 6 are converted to nucleotides using rule sequence obtained from the map $M^{(3)}$ using Algorithm 4 and are XORed with the nucleotide sequence of the DNA planes using the DNA XOR rules shown in Table 3 which gives the resultant XORed DNA planes. This algorithm is described in Algorithm 7.
- *DNA decoding:* The four chaotic DNA planes obtained from Algorithm 7 are combined back to form the cipher image by converting each nucleotide into its corresponding decimal value using decoding rules shown in Table 2. The decoding rules are decided by the chaotic sequence of 2D-LSCM $M^{(4)}$ using Algorithm 4. The algorithm for DNA Decoding is presented in Algorithm 8.

Algorithm 3: Bit Permutation

Input : Image M' and 2D-LSCM $M^{(0)}$, each of size $m \times n$
Output: Permuted Image P of size $m \times n$

```
for x ← 1 to m do
    for y ← 1 to n do
        p ← RANDOM_PERMUTE(M^(0)(x, y))          // generate random permutation from 0 to 7
        b[0...7] ← DECIMAL_TO_BINARY(M'(x, y))   // convert decimal value to 8-bit binary
        b' ← rearranged b according to permutation p
        P(x, y) ← BINARY_TO_DECIMAL(b')          // convert binary to decimal value
return P
```

Algorithm 4: Choose Rules

Input : Chaotic sequence M of dimension $m \times n$
Output: $m \times n$ sequence of rules R from 1 to 8

Initiate R as a matrix of $m \times n$
for $x \leftarrow 1$ **to** m **do**
  for $y \leftarrow 1$ **to** n **do**
    $R(x, y) \leftarrow$ `INTEGER`$(M(x, y) \times 8) + 1$ `// INTEGER(`x`) - the integer part of` x
return R

Algorithm 5: DNA Encoding

Input : Permuted image P and rule sequence R obtained from 2D-LSCM $M^{(1)}$ using Algorithm 4, each of size $m \times n$
Output: Four DNA-Plains D_1, D_2, D_3, D_4 of size $m \times n$

for $x \leftarrow 1$ **to** m **do**
  for $y \leftarrow 1$ **to** n **do**
    $r \leftarrow R(x, y)$
    $b[1...8] \leftarrow$ `DECIMAL_TO_BINARY`$(P(x, y))$
    for $i \leftarrow 1$ **to** 4 **do**
      $q_i \leftarrow 2 * b[2(i-1)] + b[2i]$
      $D_i(x, y) \leftarrow$ nucleotide according to Table 1 for quad value q_i using rule r
return D_1, D_2, D_3, D_4

Algorithm 6: DNA XOR Sequence

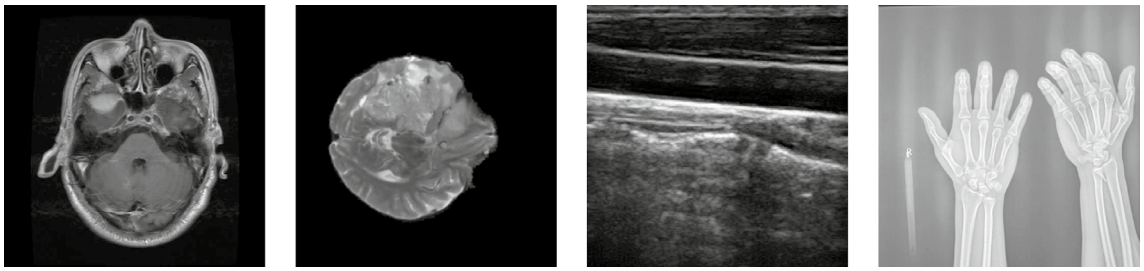
Input : 2D-LSCM $M^{(2)}$, size of image $m \times n$
Output: DNA XOR sequence P of size $m \times n$

Initialize P to size $m \times n$
for $x \leftarrow 1$ **to** $\left\lceil \frac{m \times n}{256} \right\rceil$ **do**
  $p \leftarrow$ `RANDOM_PERMUTE`$(M^{(2)}(x))$ `// generate random permutation sequence from 0 to 255`
  Append p to P
return P

Algorithm 7: DNA XOR

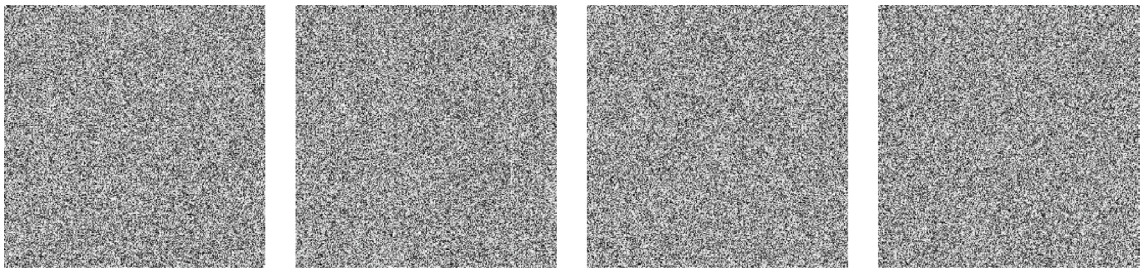
Input : DNA Plain D, DNA XOR sequence P obtained from Algorithm 6, and rule sequence R from $M^{(3)}$ obtained using Algorithm 4, each of size $m \times n$
Output: XORed DNA Plain D' of size $m \times n$

Divide DNA-Plane D into non-overlapping blocks of size 1×4
foreach *non-overlapping block* b **do**
  $x \leftarrow 1$
  foreach *element* $p \in P$ **do**
    $y \leftarrow 1$
    $r \leftarrow R(x, y)$
    $[q_1, q_2, q_3, q_4] \leftarrow$ `QUAD_VALUE` (p) `// Converting decimal to four quads (base 4)`
    $d \leftarrow$ four nucleotides according to Table 1 for quad-values $[q_1, q_2, q_3, q_4]$ using rule r
    $xor \leftarrow$ `DNA_XOR` (b,d) according to Table 3 using rule r
    Add xor to D'
    $y \leftarrow (y + 1) \bmod n$
  $x \leftarrow x + 1$
return D'



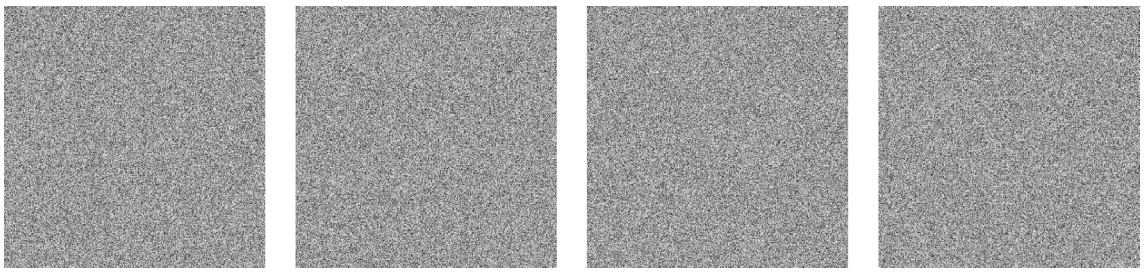
(a) CT Scan **(b)** MRI **(c)** Ultrasound **(d)** X-Ray

Fig. 7 Medical images



(a) CT Scan **(b)** MRI **(c)** Ultrasound **(d)** X-Ray

Fig. 8 Cipher images for images in Fig. 7 of dimension 256×256



(a) CT Scan **(b)** MRI **(c)** Ultrasound **(d)** X-Ray

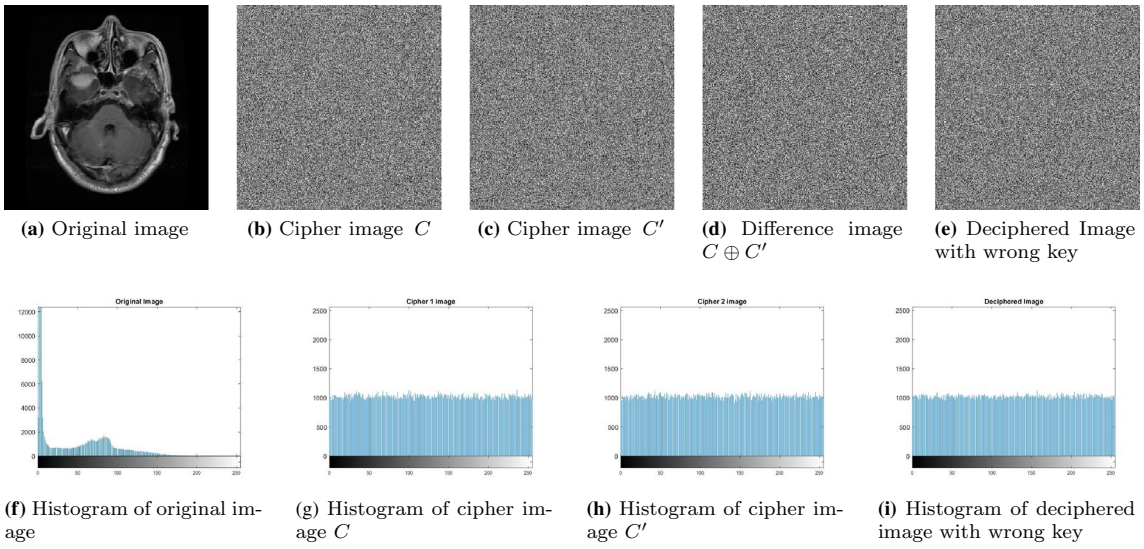
Fig. 9 Cipher images for images in Fig. 7 of dimension 512×512

Algorithm 8: DNA Decoding

Input : Four DNA Plains D'_1, D'_2, D'_3, D'_4 and rule sequence R obtained from $M^{(4)}$ using Algorithm 4, each of size $m \times n$

Output: Cipher Image C of size $m \times n$

for $x \leftarrow 1$ **to** m **do**
  for $y \leftarrow 1$ **to** n **do**
    $r \leftarrow R(x, y)$
    for $i \leftarrow 1$ **to** 4 **do**
      $q_i \leftarrow$ quad-value according to Table 2 for nucleotide $D'_i(x, y)$ using rule r
    $C(x, y) \leftarrow 64 * q_1 + 16 * q_2 + 4 * q_3 + q_4$
return C

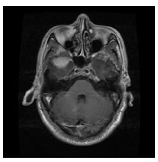
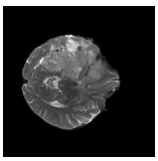
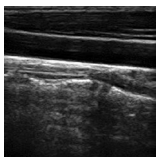


(a) Original image **(b)** Cipher image C **(c)** Cipher image C' **(d)** Difference image $C \oplus C'$ **(e)** Deciphered Image with wrong key

(f) Histogram of original image (g) Histogram of cipher image C **(h)** Histogram of cipher image C' **(i)** Histogram of deciphered image with wrong key

Fig. 10 Key sensitivity analysis

Table 4 Entropy analysis on medical images

Image								
	CT-scan		MRI		Ultrasound		X-ray	
	512×512	256×256	512×512	256×256	512×512	256×256	512×512	256×256
Original	5.391525	5.377498	3.080100	3.081431	6.852685	6.849809	6.824805	6.810973
Proposed	7.999228	7.996400	7.999327	7.996678	7.999387	7.997557	7.998562	7.994853

The overall diffusion phase uses all the Algorithms from Algorithm 3 to 8 as shown in Fig. 4. An illustration of the diffusion phase is shown in Fig. 5, where a 3×3 image is converted into a cipher image by using the various steps of the process. It can be observed that the cipher generated has no relation with either the input image or the parameters derived from the chaotic maps, which makes it very secure. The generated cipher images from this algorithm are random, secured, and resistant to various attacks. The detailed analysis is presented in the next section.

4 Experimental results and analysis

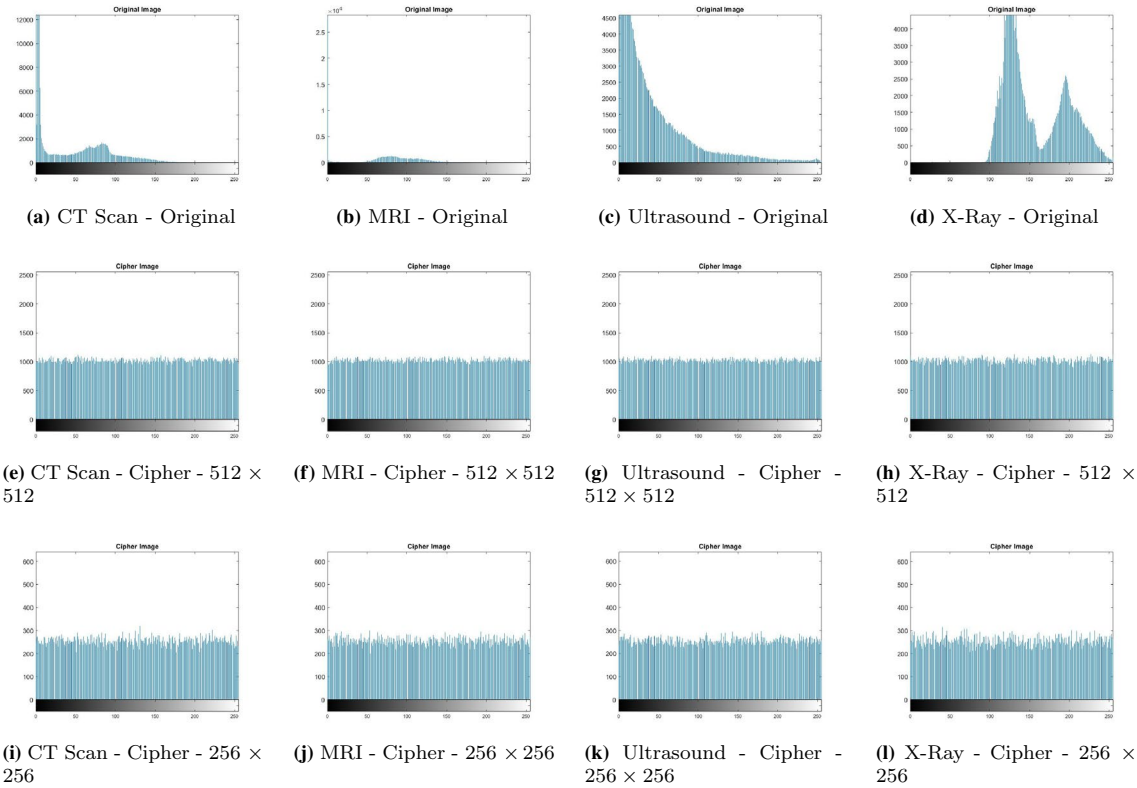
Various images have been tested by the proposed algorithm and analyzed to confirm images' security.

The four categories of medical images—CT Scan, MRI, Ultrasound, and X-Ray—as shown in Fig. 7, each of size 512×512 and 256×256 have been encoded using the encoding algorithm. The resultant cipher images are shown in Fig. 8 and 9 . These cipher images are highly random, having lost their perceptual visibility, and are noise-like. On decrypting these images using the correct key, the original image can be deciphered without any loss of information.

Security Analysis on the cryptosystem has been performed to show the robustness of the system. The system has high key-space and key sensitivity and produces cipher images having high entropy, low pixel-correlation, and looks like random noise. Furthermore, even statistical information is not preserved in the cipher image, making it impossible to decipher it without knowing the correct key. Analysis of various security parameters on the proposed cryptosystem is described as follows.

4.1 Key space

The symmetric key used for encoding and decoding must be large enough to prevent brute-force attacks by eavesdroppers. For a secured symmetric-key algorithm, the encryption key must be greater than 100 bits (Hua et al. 2018). In the

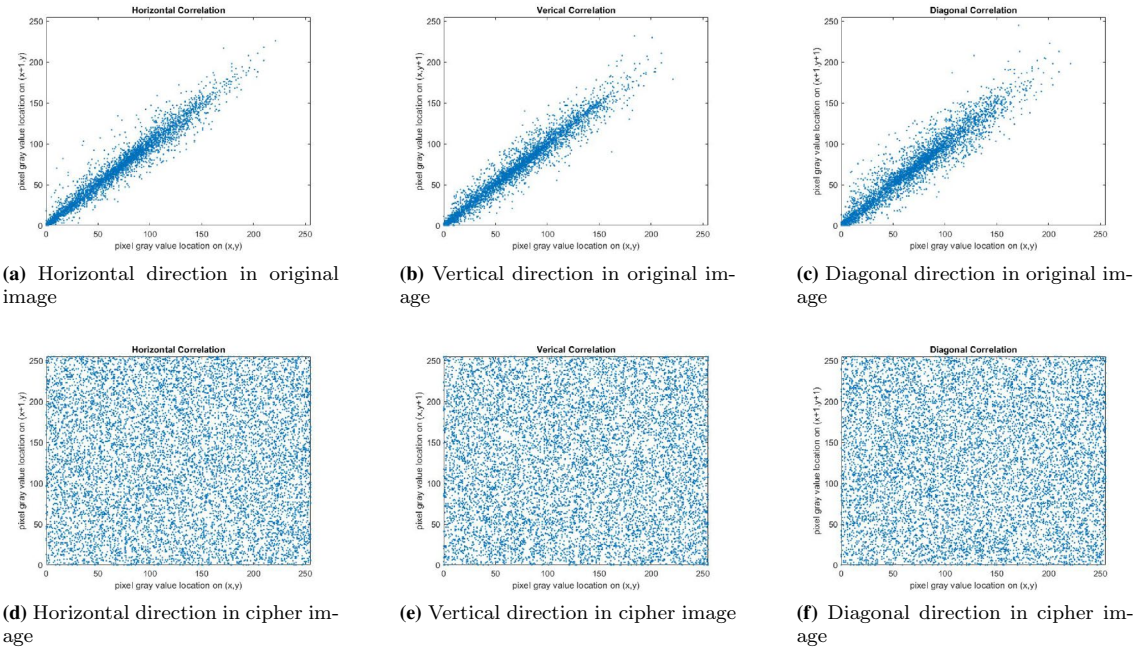


(a) CT Scan - Original **(b)** MRI - Original **(c)** Ultrasound - Original **(d)** X-Ray - Original

(e) CT Scan - Cipher - 512 × 512 **(f)** MRI - Cipher - 512 × 512 **(g)** Ultrasound - Cipher - 512 × 512 **(h)** X-Ray - Cipher - 512 × 512

(i) CT Scan - Cipher - 256 × 256 **(j)** MRI - Cipher - 256 × 256 **(k)** Ultrasound - Cipher - 256 × 256 **(l)** X-Ray - Cipher - 256 × 256

Fig. 11 Histogram analysis

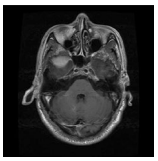
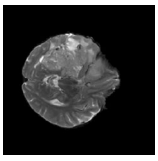
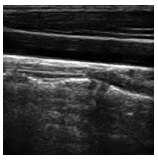


(a) Horizontal direction in original image **(b)** Vertical direction in original image **(c)** Diagonal direction in original image

(d) Horizontal direction in cipher image **(e)** Vertical direction in cipher image **(f)** Diagonal direction in cipher image

Fig. 12 Correlation between original image of CT-scan (Fig. 7a) and its cipher image (Fig. 9a) along horizontal, vertical and diagonal directions

Table 5 Correlation analysis on medical images

Image	Direction								
		CT-scan		MRI		Ultrasound		X-ray	
		512×512	256×256	512×512	256×256	512×512	256×256	512×512	256×256
Original	Horizontal	0.9904	0.9761	0.9968	0.9880	0.9830	0.9522	0.9949	0.9899
	Vertical	0.9896	0.9708	0.9937	0.9901	0.9981	0.9939	0.9829	0.9540
	Diagonal	0.9816	0.9480	0.9940	0.9782	0.9833	0.9544	0.9827	0.9551
Proposed	Horizontal	−0.0147	−0.0016	0.0094	0.0125	0.0164	0.0125	0.0139	0.0019
	Vertical	−0.0053	0.0012	−0.0104	−0.0079	−0.0059	0.0040	0.0054	−0.0096
	Diagonal	−0.0066	0.0200	0.0070	−0.0077	−0.0099	−0.0066	0.0148	−0.0136

proposed algorithm, the key size is 256-bits, which is much larger than that suggested by (Hua et al. 2018). As a result, any brute-force attack will have a complexity of 2^{256}, which is significantly high. Furthermore, each bit of the keyspace is equally essential, missing which decryption of the cipher image is nearly impossible.

4.2 Key sensitivity

Key sensitivity is defined as the amount of change in the cipher image on a very slight change of the key value. For an excellent cryptosystem, the key sensitivity should be very high. Let $C = c_0, c_1, c_2, \ldots, c_{n-1}$ represent the bits of the cipher image obtained from the original key and $C' = c'_0, c'_1, c'_2, \ldots, c'_{n-1}$ be the bits of the cipher image obtained after changing a few bits of the original key. The key sensitivity KS is measured as

$$KS = \frac{\sum_{i=0}^{n-1} c_i \oplus c'_i}{n} \times 100\% \tag{6}$$

where $\oplus$ is the bit-XOR operation. For a good cryptosystem, the key sensitivity should be around 50%.

To measure key sensitivity, the image in Fig. 7a of dimension 512×512 has been used with the key.

35 : 77 : 146 : 151 : 147 : 167 : 167 : 111 : 229 : 94 : 112 : 228 : 206 : 180 : 26 : 235 : 183 : 255 : 39 : 222 : 42 : 157 : 32 : 217 : 206 : 146 : 104 : 18 : 178 : 116 : 185 : 221.

On changing only one bit of the key, the key sensitivity was calculated to be 49.7418%. The cipher images and difference image have been shown in Fig. 10.

4.3 Statistical attacks

An encryption algorithm for images, where data is highly redundant, needs to preserve perceptual security and prevent statistical attacks. Furthermore, the cipher images must be highly random and uncorrelated, giving no clue what the original image may be. The statistical analysis for the proposed cryptosystem is provided in the following subsections:

4.3.1 Entropy analysis

Entropy is a measure of randomness. For a source having N symbols each with the probability of occurrence $p_i, i = 1, 2, \ldots, N$, it is defined by (Shannon 1948) as

$$H(S) = \sum_{i=1}^{N} p_i \log_2 \frac{1}{p_i} \tag{7}$$

The higher the entropy, the higher is the randomness in the data. Shannon (1948) proposed the limits for entropy as

$$0 \leq H(S) \leq \log_2 N \tag{8}$$

For a grayscale image where every pixel is represented in 8-bits, the total number of possible symbols is $N = 2^8$. Therefore, the entropy can have a maximum value of 8 bits/symbol. The entropies of the images in Fig. 7 and their cipher images Fig. 8 and 9 are summarized in the Table 4. All the cipher images' entropies are near to 8 bits/symbol, proving their high randomness.

4.3.2 Histogram analysis

The histogram gives statistical information about the image. A good cipher image must have an equalized and

well-distributed histogram to not preserve the statistical information of the source. The histograms for the original images in Fig. 7 and their corresponding cipher images in Fig. 8 and 9 are shown in Fig.11.

It is evident from the histograms that the cipher image's statistical information is completely lost, and the original image cannot be traced back or guessed by mere histogram analysis.

4.3.3 Correlation analysis

In an image, the pixels are highly correlated. Therefore, a good cryptosystem should remove this correlation, and the cipher image should be highly uncorrelated. Karl Pearson gave a measurement for correlation (Benesty et al. 2009) as

Table 6 NPCR and UACI analysis on medical images

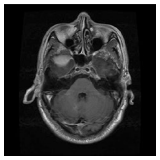
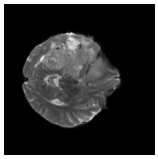
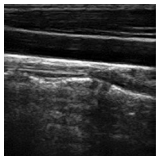
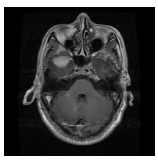
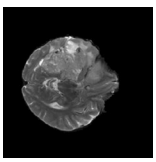
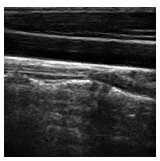
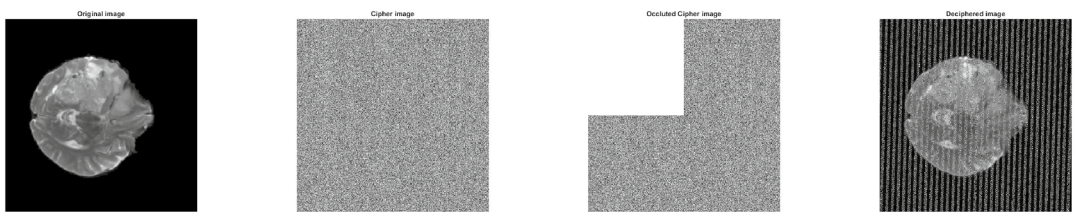
Image								
	CT-scan		MRI		Ultrasound		X-ray	
	512×512	256×256	512×512	256×256	512×512	256×256	512×512	256×256
NPCR	99.6017	99.5544	99.6086	99.5316	99.6174	99.5789	99.5667	99.5956
UACI	0.3345	0.3331	0.3339	0.3348	0.3347	0.3344	0.3324	0.3351

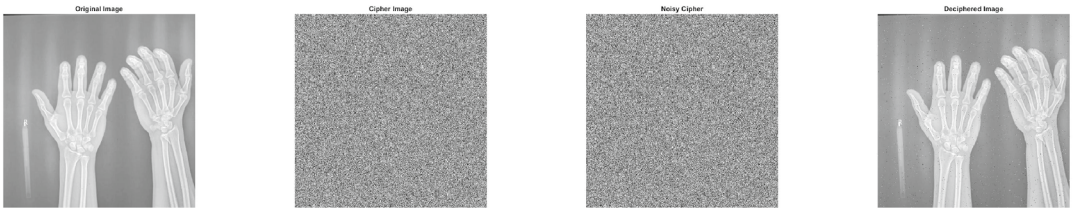
Table 7 Cross-correlation analysis on medical images

Image								
	CT-scan		MRI		Ultrasound		X-ray	
	512×512	256×256	512×512	256×256	512×512	256×256	512×512	256×256
Cross-correlation	0.0069	−0.0122	−0.0209	0.0034	−0.0127	−0.0140	0.0014	0.0041



(a) Original Image **(b)** Cipher Image **(c)** Occluted cipher image with one-fourth of information lost **(d)** Deciphered image from the occluted cipher

Fig. 13 Occlusion attack on cipher image



(a) Original image **(b)** Cipher image **(c)** Noisy cipher image with salt and pepper noise **(d)** Deciphered image from the noisy cipher

Fig. 14 Noise attack on cipher image

$$\rho(x,y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{9}$$

Where, x and y are two vectors of size n, and $\bar{x}$ and $\bar{y}$ represent their mean values, respectively.

The value of ρ lies in the range $[-1, 1]$. Thus, a value close to 0 indicates no correlation, whereas a value close to 1 indicates a very high correlation.

Correlation for images is measured horizontally, vertically, and diagonally. The plots for correlation between original and cipher images for 10,000 randomly chosen points of CT-scan (512×512) in Fig. 7a is shown in Fig. 12. The correlation coefficients of different images and their cipher images are summarized in Table 5. It can be inferred from Table 5 that the correlation coefficients along all three directions for the cipher image are close to zero. Therefore, they are highly random and have no correlation among adjacent pixels, contrary to the plain images, which have high correlation values near one.

4.4 Differential attack

4.4.1 NPCR and UACI analysis

Differential cryptanalysis is a powerful attack on cryptosystems which was first described by Biham and Shamir (1991). For image cryptosystems, Number of Pixels Changing Rate

Table 8 Analysis of different steps of the proposed cryptosystem

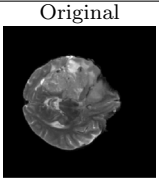
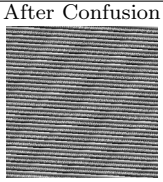
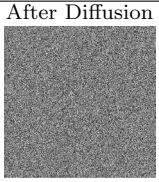
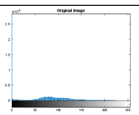
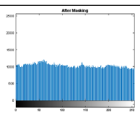
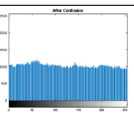
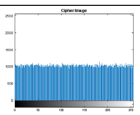
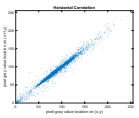
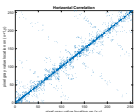
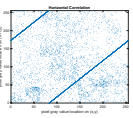
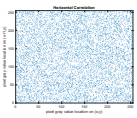
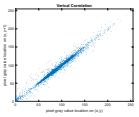
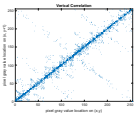
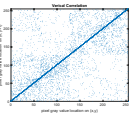
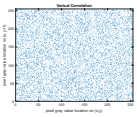
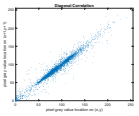
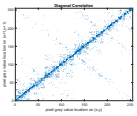
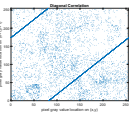
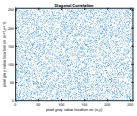
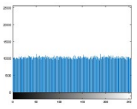
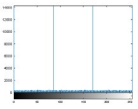
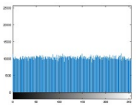
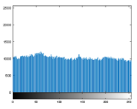
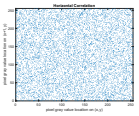
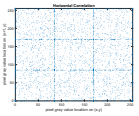
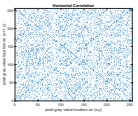
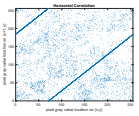
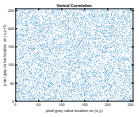
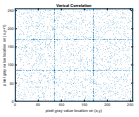
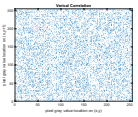
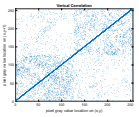
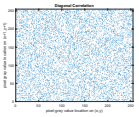
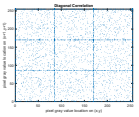
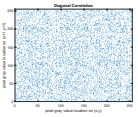
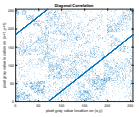
Image		Original	After Masking	After Confusion	After Diffusion
					
Histogram					
Entropy		3.0801	7.9928	7.9928	7.9993
Correlation	Horizontal				
		0.9970	0.9784	-0.2251	0.0094
	Vertical				
		0.9973	0.9479	0.8181	-0.0104
	Diagonal				
		0.9942	0.9416	-0.2196	0.0200

Table 9 Analysis of cipher images by eliminating one of the steps of the proposed cryptosystem

Image		Proposed Method	Without Masking	Without Confusion	Without Diffusion
					
Histogram					
Entropy		7.9993	4.7027	7.9986	7.9979
Correlation	Horizontal	 0.0094	 -0.0199	 0.0005	 -0.1293
	Vertical	 -0.0109	 -0.0061	 0.0158	 0.7522
	Diagonal	 0.0200	 0.0046	 0.0115	 -0.0905

(NPCR) and Unified Averaged Changed Intensity (UACI) are two common parameters that check the resistance of the system to differential attacks.

Suppose an image $\mathcal{P}$ is encrypted using a cryptosystem. Let $\mathcal{C}$ be the cipher image obtained using a key $\mathcal{K}$. One pixel of the image $\mathcal{P}$ is changed to obtain the image $\mathcal{P}'$ and is encrypted using the same key $\mathcal{K}$ to obtain the cipher image $\mathcal{C}'$. For an image of size $m \times n$ of depth p bits per pixel (Wu et al. 2011):

$$\mathcal{D}(x,y) = \begin{cases} 1, & \text{if } \mathcal{C}(x,y) \neq \mathcal{C}'(x,y) \\ 0, & \text{if } \mathcal{C}(x,y) = \mathcal{C}'(x,y) \end{cases} \tag{10}$$

$$\text{NPCR} : \mathcal{N}(\mathcal{C},\mathcal{C}') = \sum_{x,y} \frac{\mathcal{D}(x,y)}{m \times n} \times 100\% \tag{11}$$

$$\text{UACI} : \mathcal{U}(\mathcal{C},\mathcal{C}') = \sum_{x,y} \frac{|\mathcal{C}(x,y) - \mathcal{C}'(x,y)|}{(2^p - 1) \cdot (m \times n)} \tag{12}$$

where $x = 1, 2, \ldots, m$; $y = 1, 2, \ldots, n$ and $|a|$ denotes the absolute value of a. For a grayscale image of 8-bit depth, $p = 8$. The NPCR and UACI for different images are shown in Table 6.

4.4.2 Cross-correlation analysis

A good encrypted image should have a low correlation among the original and cipher images' corresponding pixels. A low cross-correlation ensures that differential attack can be resisted by removing similarity among the corresponding pixels while generating the cipher image. In the proposed algorithm, this is removed by the confusion and diffusion process. Ten thousand random pixels have been chosen from the original image and their corresponding cipher images to form two vectors x and y to measure the cross-correlation. Then the coefficient of cross-correlation is calculated using the formula mentioned in Eq. 9.

The results of cross-correlation obtained for the medical images have been summarized in Table 7. It can be seen that the values are all close to 0, indicating that there is a very low correlation among the corresponding pixels of the original and the cipher images.

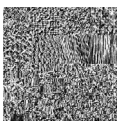
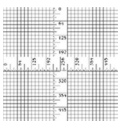
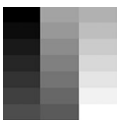
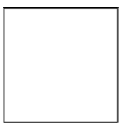
4.5 Occlusion and noise attack

In any communication channel, there is always a possibility of a loss of information due to the presence of noise or other adversaries. However, the decryption algorithm can still

Table 10 Security analysis on natural images, each of size 512 × 512

Image			 Boat	 Baboon	 Camera-man	 Jetplane	 Lena	 Pepper
Entropy	Original		7.123758	7.292549	7.047955	6.702463	7.445061	7.571478
	Cipher		7.998973	7.998621	7.999230	7.999136	7.998090	7.999370
Correlation	Original	Horizontal	0.9802	0.9209	0.9897	0.9715	0.9860	0.9852
		Vertical	0.9692	0.9336	0.9802	0.9699	0.9736	0.9830
		Diagonal	0.9529	0.8753	0.9719	0.9475	0.9640	0.9749
	Cipher	Horizontal	−0.0119	0.0172	0.0073	0.0172	0.0129	−0.0181
		Vertical	0.0047	−0.0064	0.0001	−0.0003	−0.0002	−0.0176
		Diagonal	0.0066	−0.0042	0.0051	0.0131	−0.0024	0.0002
NPCR			99.5998	99.6136	99.6086	99.6201	99.5892	99.6078
UACI			0.3355	0.3352	0.3348	0.3350	0.3350	0.3348
Cross-correlation			0.0176	−0.0063	0.0161	0.0172	−0.0111	0.0121

Table 11 Security analysis on homogeneous and textured pattern, each of size 512 × 512

Image			 Brick wall	 Mosaic	 Ruler	 Wedge	 All White	 All black
Entropy	Original		6.7899	7.9117	0.5	4.3922	0	0
	Cipher		7.9987	7.9993	7.9992	7.9982	7.999291	7.999291
Correlation	Original	Horizontal	0.9192	0.8114	0.6076	0.9997	–	–
		Vertical	0.8290	0.6867	0.4167	0.9915	–	–
		Diagonal	0.8050	0.5742	−0.0198	0.9912	–	–
	Cipher	Horizontal	0.0081	0.0044	−0.0171	−0.0023	0.0109	0.0109
		Vertical	−0.0083	0.0023	0.0011	−0.0001	0.0186	0.0186
		Diagonal	−0.0173	0.0033	0.0020	−0.0038	0.0203	0.0203
NPCR			99.5975	99.6124	99.8913	99.5232	99.6093	99.6094
UACI			0.3349	0.3342	0.3469	0.331	0.3348	0.3348
Cross-correlation			−0.0148	0.0061	−0.0157	−0.0203	–	–

recover some part of the image even if it does not receive the entire cipher image.

In an occlusion attack, a significant part of the cipher image is lost during transmission. The decryption algorithm tries to recover the original image from it using the same key. As a result, some of the information from the recovered image may be lost. However, it may retain most of the visual information to get an idea of the original image. Figure 13 shows the effect of occlusion attack on the cipher image generated by the proposed algorithm. One-fourth of the cipher image is lost during transmission. Yet, the decryption algorithm can retain back some of the visual information from the image, which is good enough to understand the original image's visual content.

In noise attack, some of the cipher images' pixel values are changed during transmission due to errors in the channel. Figure 14 shows the effect of noise attack where the cipher image obtained from the encryption algorithm is added with salt-and-pepper-noise. When this noisy cipher is decrypted, the resultant deciphered image still holds the original image's visual information, and therefore, the cryptosystem resists the noise attack.

Table 12 Comparative analysis on MRI image of dimension 256×256

Method	Entropy	Correlation coefficient			NPCR	UACI
		Horizontal	Vertical	Diagonal		
Proposed	7.9967	0.0125	−0.0079	−0.0077	0.9953	0.3348
Dagadu et al. (2019)	7.9969	−0.0037	−0.0017	0.0009	0.9962	0.3353
Barik and Changder (2020)	7.9969	−0.0205	−0.0492	0.0391	0.9531	0.3952
Dridi et al. (2016)	7.9951	0.0015	0.0026	0.0011	0.9551	0.3342

Table 13 Comparative analysis on baboon image of dimension 256×256

Method	Entropy	Correlation coefficient			NPCR	UACI
		Horizontal	Vertical	Diagonal		
Proposed	7.9973	0.0092	−0.0102	−0.0084	0.9958	0.3340
Patel et al. (2020)	7.9893	−0.0010	−0.0005	0.0007	0.9960	0.2739
El-Khamy and Mohamed (2021)	7.9995	0.0012	0.0001	0.0011	0.9965	0.3354
Zhang et al. (2018)	7.9973	0.0016	−0.0014	−0.0047	0.9958	0.3342

4.6 Analysis of different steps of the cryptosystem

The importance of the various steps of the proposed cryptosystem is discussed in this section. Table 8 shows the parameters of an image after various steps of the proposed method. The original image has very low entropy and high inter-pixel correlation. Also, the distribution of the pixel intensities is highly non-uniform, as shown in the histogram. The masking step increases the entropy, however, the correlation is still high, and the image is very similar to the original image. The confusion step removes the visual information of the image, yet the resultant confused images have good amount of inter-pixel correlation. This correlation is removed by the diffusion process, which results in a cipher having high entropy, equalized histogram, and a low coefficient of correlation.

Table 9 shows the various parameters of the ciphers obtained by eliminating one of the steps of the proposed cryptosystem. The cipher image obtained without masking has very low entropy, the one without confusion has visual similarities, and the one without diffusion has a high inter-pixel correlation coefficient. It shows that all three steps are essential for the generation of a secured cipher image.

4.7 On natural images

The proposed algorithm has been tested on natural images, commonly used in literature to test different cryptosystems parameters. Results show that the proposed work is equally suitable for these images as well. Furthermore, deciphered images are lossless, having the same values as those of the original images. Different parameters for natural images are summarized in Table 10.

4.8 On textures and homogeneous patterns

The proposed cryptosystem has also been tested on textures and homogeneous patterns like all white and all black. All these images contain a wide variety of characteristics. For example, the textured images contain repetitive patterns, whereas the homogeneous ones contain all the pixels of the same intensity. The properties of the resultant cipher of all these images are summarized in Table 11. All the values obtained from the analysis of these images meet the criteria of a secured image cryptosystem.

4.9 Time and space complexity analysis

There are three significant steps in the proposed algorithm, namely, masking, confusion, and diffusion. For an image of dimension $n \times n$, the resultant cipher is also of the same size. The masking step traverses each element once and hence takes a time of $O(n^2)$. The confusion step works on all the pixels of the image and is iterated over p number of times, where p is the largest prime which is less than or equal to $n/2$. Asymptotically, this step takes a time of $O(pn^2) \approx O(n^3)$. The diffusion step performs bit-level shuffling, which takes time $O(4 \times n^2)$, the DNA encoding step takes $O(n^2)$, the DNA XOR step takes $4 \times O(n^2)$, and the DNA decoding also takes $O(n^2)$. Overall, the time complexity of the diffusion process is bound by $O(n^2)$. Therefore the overall time complexity of the encryption algorithm turns out to be $O(n^2 + n^3 + n^2) = O(n^3)$.

The various maps used for generating pseudo-random numbers, the sequences for choosing the rules, and performing various operations like DNA XOR have a size equal to

Table 14 Comparative analysis on the entire dataset with Ref. (Dagadu et al. 2019)

Image	Method	Entropy	Correlation coefficient			NPCR	UACI	Cross
			Horizontal	Vertical	Diagonal			Correlation
CT-scan	Proposed	7.9964	−0.0016	0.0012	0.0200	0.9955	0.3331	−0.0122
256 × 256	Ref.	7.9972	−0.0016	0.0043	− 0.0061	0.9964	0.3343	0.0012
CT-scan	Proposed	7.9992	−0.0147	−0.0053	−0.0066	0.9960	0.3345	0.0069
512 × 512	Ref.	7.9993	−0.0035	0.0019	−0.0031	0.9961	0.3351	−0.0038
MRI	Proposed	7.9967	0.0125	−0.0079	−0.0077	0.9953	0.3348	0.0034
256 × 256	Ref.	7.9969	−0.0037	−0.0017	0.0009	0.9962	0.3353	−0.0017
MRI	Proposed	7.9993	0.0094	−0.0104	0.0070	0.9960	0.3339	−0.0209
512 × 512	Ref.	7.9993	− 0.0009	0.0007	0.0004	0.9962	0.3353	0.0166
Ultrasound	Proposed	7.9975	0.0125	0.0040	−0.0066	0.9957	0.3344	−0.0140
256 × 256	Ref.	7.9972	−0.0025	−0.0011	−0.0014	0.9960	0.3364	0.0090
Ultrasound	Proposed	7.9993	0.0164	−0.0059	−0.0099	0.9961	0.3347	−0.0127
512 × 512	Ref.	7.9993	− 0.0022	0.0028	− 0.0016	0.9960	0.3351	0.0074
X-ray	Proposed	7.9948	0.0019	−0.0096	−0.0136	0.9959	0.3351	0.0041
256 × 256	Ref.	7.9970	0.0022	− 0.0034	0.0043	0.9960	0.3351	−0.0084
X-ray	Proposed	7.9985	0.0139	0.0054	0.0148	0.9956	0.3324	0.0014
512 × 512	Ref.	7.9993	0.0011	−0.0005	0.0002	0.9961	0.3348	−0.0122
Boat	Proposed	7.9989	−0.0119	0.0047	0.0066	0.9959	0.3355	0.0176
512 × 512	Ref.	7.9916	−0.0103	0.0066	−0.0036	0.9962	0.3351	−0.0045
Baboon	Proposed	7.9986	0.0172	−0.0064	−0.0042	0.9961	0.3352	−0.0063
512 × 512	Ref.	7.9959	0.0162	−0.0047	0.0060	0.9960	0.3338	0.0026
Cameraman	Proposed	7.9992	0.0073	0.0001	0.0051	0.9960	0.3348	0.0161
512 × 512	Ref.	7.9631	−0.0053	0.0001	−0.0001	0.9963	0.3378	−0.0038
Jetplane	Proposed	7.9991	0.0172	−0.0003	0.0131	0.9962	0.3350	0.0172
512 × 512	Ref.	7.9992	0.0190	−0.0054	0.0050	0.9959	0.3344	0.0057
Lena	Proposed	7.9980	0.0129	−0.0002	− 0.0024	0.9958	0.3350	−0.0111
512 × 512	Ref.	7.9994	−0.0034	−0.0115	0.0013	0.9960	0.3289	0.0086
Pepper	Proposed	7.9993	−0.0181	−0.0176	0.0002	0.9960	0.3348	0.0121
512 × 512	Ref.	7.9909	−0.0074	0.0111	−0.0147	0.4983	0.1677	0.0220
Brick wall	Proposed	7.9987	0.0081	−0.0083	−0.0173	0.9959	0.3349	−0.0148
512 × 512	Ref.	7.9859	−0.0018	−0.0004	0.0084	0.4982	0.1672	0.0080
Mosaic	Proposed	7.9993	0.0044	0.0023	0.0033	0.9961	0.3342	0.0061
512 × 512	Ref.	7.9993	−0.0057	−0.0047	−0.0158	0.9962	0.3345	0.0113
Ruler	Proposed	7.9992	0.0171	0.0011	0.0020	0.9989	0.3469	−0.0157
512 × 512	Ref.	7.9993	0.0104	−0.0068	0.0048	0.9963	0.3353	−0.0312
Wedge	Proposed	7.9982	−0.0023	−0.0001	−0.0038	0.9952	0.3310	−0.0203
512 × 512	Ref.	7.9992	−0.0057	−0.0065	−0.0010	0.9960	0.3365	−0.0071
All-white	Proposed	7.9992	0.0109	0.0186	0.0203	0.9960	0.3348	–
512 × 512	Ref.	5.9397	0.0217	−0.0168	−0.0007	0.9962	0.3623	–
All-black	Proposed	7.9992	0.0109	0.0186	0.0203	0.9960	0.3348	–
512 × 512	Ref.	5.9420	0.0008	−0.0053	0.0180	0.9961	0.3621	–

the size of the image. All the non-DNA-based operations can be done in the image matrix itself. The DNA-based operations require four DNA planes, each containing DNA nucleotides of the same dimension as the original image. Therefore, the space complexity of the proposed algorithm is $O(n^2)$.

5 Comparative analysis

The algorithm's performance has been compared with some existing algorithms on the encryption of medical and natural images. It is evident from the data that the proposed algorithm meets the criteria for a robust cryptosystem.

MRI image of size 256×256 has been used for comparison, whose results are summarized in Table 12. For natural images, the Baboon image of size 256×256 has been compared with existing algorithms, the results of which are shown in Table 13.

The performance of the proposed algorithm has also been compared with Dagadu et al. (2019) on the entire data set. It can be seen that the values obtained from the proposed method and that proposed by Dagadu et al. (2019) are comparable for all the parameters. It shows that the proposed cryptosystem is at par with the state-of-the-art methods. Moreover, the proposed method performs much better for homogeneous images like all white and all black, for which the method of Dagadu et al. (2019) fails to give a proper entropy. The results are shown in Table 14.

6 Conclusion

This paper presents a novel algorithm for encrypting medical images, which works equally well for natural and textured images. The algorithm proposes a novel masking method, which is keyless. It is applied before the confusion and diffusion process to increase the image's entropy. This method is very effective for generating a high-quality cipher image, especially when the original images have a large homogeneous content, which is seen in most of the medical images. The confusion step uses ACM, and the number of iterations has been chosen in such a way as to obtain a highly scrambled image and avoid periodicity. The diffusion algorithm is the most important step of the cryptosystem, which changes the pixel values of the image and acts on both - bit level and pixel level. It incorporates the eight different rules for DNA-encoding, DNA-XOR, and DNA-decoding and generates the final cipher image, which is random noise-like and has a very low correlation with the original images.

The proposed algorithm has been tested on several security parameters for a variety of images ranging from medical images like CT-SCAN, MRI, Ultrasound, X-RAY to a number of natural images commonly used in literature. Many texture and homogeneous patterns have also been used to test the quality and security of the proposed algorithm. The cipher images obtained have high entropy, a low inter-pixel, cross-correlation, NPCR and UACI in the acceptable range. Moreover, the images can also resist noise and occlusion attacks. The parameters of key size and key sensitivity also ensure that the proposed algorithm meets the criteria of a secure image cryptosystem.

The field of DNA-based image cryptography is a new and interesting area of research. This paper presents some of the operations of DNA-based cryptography that can be used to generate high-quality cipher images for secure transmission. Researchers can further explore it and devise new confusion and diffusion methods. New DNA-based operations can be designed for the generation of more effective cryptosystems. Researchers can also explore the possibilities of different formulas and techniques of masking and design faster and more secure algorithms for encryption and decryption of medical and natural images.

References

Akkasaligar PT, Biradar S (2016) Secure medical image encryption based on intensity level using chao's theory and DNA cryptography. In: 2016 IEEE international conference on computational intelligence and computing research (ICCIC), IEEE, pp 1–6, https://doi.org/10.1109/ICCIC.2016.7919681

Akkasaligar PT, Biradar S (2018) Medical image encryption with integrity using DNA and chaotic map. In: International conference on recent trends in image processing and pattern recognition, Springer, pp 143–153

Akkasaligar PT, Biradar S (2020) Selective medical image encryption using DNA cryptography. Inf Secur J Glob Perspect 29(2):91–101. https://doi.org/10.1080/19393555.2020.1718248

Arnol'd VI, Avez A (1968) Ergodic problems of classical mechanics http://cds.cern.ch/record/1987366

Barik RC, Changder S (2020) A novel and efficient amino acid codon based medical image encryption scheme colligating multiple chaotic maps. Multimed Tools Appl. https://doi.org/10.1007/s11042-020-09930-2

Belazi A, Talha M, Kharbech S, Xiang W (2019) Novel medical image encryption scheme based on chaos and DNA encoding. IEEE Access 7:36667–36681. https://doi.org/10.1109/ACCESS.2019.2906292

Benesty J, Chen J, Huang Y, Cohen I (2009) Pearson correlation coefficient. Noise reduction in speech processing. Springer, Berlin, pp 1–4. https://doi.org/10.1007/978-3-642-00296-0

Biham E, Shamir A (1991) Differential cryptanalysis of DES-like cryptosystems. J Cryptol 4(1):3–72. https://doi.org/10.1007/BF00630563

Chen J, Zhu Z, Zhang LB, Zhang Y, Yang BQ (2018) Exploiting self-adaptive permutation-diffusion and DNA random encoding for secure and efficient image encryption. Signal Process 142:340–353. https://doi.org/10.1016/j.sigpro.2017.07.034

Dagadu JC, Li JP, Aboagye EO (2019) Medical image encryption based on hybrid chaotic DNA diffusion. Wirel Pers Commun 108(1):591–612. https://doi.org/10.1007/s11277-019-06420-z

Dridi M, Hajjaji MA, Bouallegue B, Mtibaa A (2016) Cryptography of medical images based on a combination between chaotic and neural network. IET Image Process 10(11):830–839. https://doi.org/10.1049/iet-ipr.2015.0868

Dua M, Wesanekar A, Gupta V, Bhola M, Dua S (2019) Differential evolution optimization of intertwining logistic map-DNA based image encryption technique. J Ambient Intell Hum Comput. https://doi.org/10.1007/s12652-019-01580-z

El-Khamy SE, Mohamed AG (2021) An efficient DNA-inspired image encryption algorithm based on hyper-chaotic maps and wavelet fusion. Multimed Tools Appl. https://doi.org/10.1007/s11042-021-10527-6

Fridrich J (1997) Image encryption based on chaotic maps. In: 1997 IEEE international conference on systems, man, and cybernetics. Computational cybernetics and simulation, IEEE, vol 2, pp. 1105–1110. https://doi.org/10.1109/ICSMC.1997.638097

Haskins G, Kruger U, Yan P (2020) Deep learning in medical image registration: a survey. Mach Vis Appl 31(1):1–18. https://doi.org/10.1007/s00138-020-01060-x

Hua Z, Jin F, Xu B, Huang H (2018) 2D logistic-sine-coupling map for image encryption. Signal Process 149:148–161. https://doi.org/10.1016/j.sigpro.2018.03.010

Jeevitha S, Prabha NA (2020) Novel medical image encryption using DWT block-based scrambling and edge maps. J Ambient Intell Hum Comput. https://doi.org/10.1007/s12652-020-02399-9

Kari L, Mahalingam K (2007) Watson-crick conjugate and commutative words. In: International workshop on DNA-based computers. Springer, pp 273–283. https://doi.org/10.1007/978-3-540-77962-929

Khare P, Srivastava VK (2020) A secured and robust medical image watermarking approach for protecting integrity of medical images. Trans Emerg Telecommun Technol 32:e3918. https://doi.org/10.1002/ett.3918

Li W, Feng C, Yu K, Zhao D (2020) Miss-d: a fast and scalable framework of medical image storage service based on distributed file system. Comput Methods Prog Biomed. https://doi.org/10.1016/j.cmpb.2019.105189

Parameshachari B, Panduranga H, Naveenkumar S et al. (2017) Partial encryption of medical images by dual DNA addition using DNA encoding. In: 2017 International conference on recent innovations in signal processing and embedded systems (RISE), IEEE, pp 310–314. https://doi.org/10.1109/RISE.2017.8378172

Parvees MM, Samath JA, Bose BP (2016) Secured medical images-a chaotic pixel scrambling approach. J Med Syst 40(11):232. https://doi.org/10.1007/s10916-016-0611-5

Patel S, Bharath K, Kumar R (2020) Symmetric keys image encryption and decryption using 3d chaotic maps with DNA encoding technique. Multimed Tools Appl 79(43):31739–31757. https://doi.org/10.1007/s11042-020-09551-9

Połap D (2019) Analysis of skin marks through the use of intelligent things. IEEE Access 7:149355–149363. https://doi.org/10.1109/ACCESS.2019.2947354

Połap D (2020) An adaptive genetic algorithm as a supporting mechanism for microscopy image analysis in a cascade of convolution neural networks. Appl Soft Comput 97:106824. https://doi.org/10.1016/j.asoc.2020.106824

Połap D, Srivastava G (2020) Neural image reconstruction using a heuristic validation mechanism. Neural Comput Appl. https://doi.org/10.1007/s00521-020-05046-8

Qiao Z, El Assad S, Taralova I (2020) Design of secure cryptosystem based on chaotic components and AES S-box. AEU Int J Electron Commun. https://doi.org/10.1016/j.aeue.2020.153205

Rijmen V, Daemen J (2001) Advanced encryption standard. Proceedings of federal information processing standards publications, National Institute of Standards and Technology pp 19–22, https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf

Shannon CE (1948) A mathematical theory of communication. Bell Syst Tech J 27(3):379–423. https://doi.org/10.1002/j.1538-7305.1948.tb01338.x

Smid ME, Branstad DK (1988) Data encryption standard: past and future. Proc IEEE 76(5):550–559. https://doi.org/10.1109/5.4441

Stalin S, Maheshwary P, Shukla PK, Maheshwari M, Gour B, Khare A (2019) Fast and secure medical image encryption based on non linear 4D logistic map and DNA sequences (nl4dlm_dna). J Med Syst 43(8):267. https://doi.org/10.1007/s10916-019-1389-z

Stevenson D (1981) A proposed standard for binary floating-point arithmetic. Computer 14(03):51–62. https://doi.org/10.1109/C-M.1981.220377

Tsafack N, Sankar S, Abd-El-Atty B, Kengne J, Jithin K, Belazi A, Mehmood I, Bashir AK, Song OY, Abd El-Latif AA (2020) A new chaotic map with dynamic analysis and encryption application in internet of health things. IEEE Access 8:137731–137744. https://doi.org/10.1109/ACCESS.2020.3010794

Wang Y, Kung L, Byrd TA (2018a) Big data analytics: understanding its capabilities and potential benefits for healthcare organizations. Technol Forecast Soc Change 126:3–13. https://doi.org/10.1016/j.techfore.2015.12.019

Wang Y, Zhao Y, Zhou Q, Lin Z (2018b) Image encryption using partitioned cellular automata. Neurocomputing 275:1318–1332. https://doi.org/10.1016/j.neucom.2017.09.068

Wu Y, Noonan JP, Agaian S et al (2011) NPCR and UACI randomness tests for image encryption. Cyber journals: multidisciplinary journals in science and technology. J Sel Areas Telecommun (JSAT) 1(2):31–38

Wu J, Liao X, Yang B (2017) Color image encryption based on chaotic systems and elliptic curve ElGamal scheme. Signal Process 141:109–124. https://doi.org/10.1016/j.sigpro.2017.04.006

Zhang W, Yu H, Zhu Zl et al (2018) An image encryption scheme using self-adaptive selective permutation and inter-intra-block feedback diffusion. Signal Process 151:130–143. https://doi.org/10.1016/j.sigpro.2018.05.008