Computer Modeling in
Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2024.054976

ARTICLE

AI-Powered Image Security: Utilizing Autoencoders for Advanced Medical Image Encryption

Fehaid Alqahtani*

Department of Computer Science, King Fahad Naval Academy, Al Jubail, 35512, Saudi Arabia

*Corresponding Author: Fehaid Alqahtani. Email: f-alqahtani@rsnf.gov.sa

Received: 13 June 2024 Accepted: 15 August 2024 Published: 27 September 2024

ABSTRACT

With the rapid advancement in artificial intelligence (AI) and its application in the Internet of Things (IoT), intelligent technologies are being introduced in the medical field, giving rise to smart healthcare systems. The medical imaging data contains sensitive information, which can easily be stolen or tampered with, necessitating secure encryption schemes designed specifically to protect these images. This paper introduces an artificial intelligence-driven novel encryption scheme tailored for the secure transmission and storage of high-resolution medical images. The proposed scheme utilizes an artificial intelligence-based autoencoder to compress high-resolution medical images and to facilitate fast encryption and decryption. The proposed autoencoder retains important diagnostic information even after reducing the image dimensions. The low-resolution images then undergo a four-stage encryption process. The first two encryption stages involve permutation and the next two stages involve confusion. The first two stages ensure the disruption of the structure of the image, making it secure against statistical attacks. Whereas the two stages of confusion ensure the effective concealment of the pixel values making it difficult to decrypt without secret keys. This encrypted image is then safe for storage or transmission. The proposed scheme has been extensively evaluated against various attacks and statistical security parameters confirming its effectiveness in securing medical image data.

KEYWORDS

Artificial Intelligence; image encryption; chaos; medical image encryption

1 Introduction

In this digital age, with the widespread use of smart healthcare and electronic health record systems, securing medical images has become critically important. Medical images that contain private information about patients can be attacked, modified, and tampered with [1]. To protect the privacy of the patients, the information in these images should be effectively secured [2]. Image encryption algorithms can come in handy in this regard. Image encryption converts the images into a format that conceals the information and protects it from cyber-attacks and unauthorized access [3,4]. An efficient image encryption algorithm should have two most important properties, i.e., confusion and permutation [5,6]. The confusion part of the encryption scheme focuses on changing the values of the pixels and the permutation part of the scheme focuses on mixing the positions of the pixels. A carefully designed encryption scheme consisting of these two properties can effectively secure medical images [7,8].

Recently, artificial intelligence has been used in cryptography in many ways. Artificial intelligence-based autoencoders are extensively being used in image encryption applications. One of the important applications of autoencoders is using them as a dimensionality reduction tool [9]. Autoencoders are type of neural network that encode the large-size input images to their low-size equivalent images. By design, these networks learn to compress data (encode) into a lower-dimensional space and then reconstruct it back (decode) to its original form, with as little loss of information as possible. This process of encoding followed by decoding helps to discover and preserve the most important features in the data. Autoencoders are particularly effective because they are trained to minimize the reconstruction error, encouraging the model to capture and prioritize the most salient features of the data. This ability to reduce dimensionality while retaining significant information makes autoencoders valuable tools in areas like image processing, where reducing the size of data can significantly enhance computational efficiency without sacrificing critical content. When designing an encryption algorithm, it is important to ensure that, in addition to being secure, it encrypts large images efficiently and with good encryption speeds [10]. Hence, the motivation of this paper is to design an image encryption scheme that protects medical images with high security while efficiently handling large high-definition (HD) images and providing fast encryption speed.

This paper introduces a novel convolutional autoencoder-based image encryption scheme designed specifically for medical images. The proposed architecture of the convolutional autoencoder converts HD medical images into their low-resolution equivalents but keeps all the important information. These low-resolution equivalents are then encrypted and transmitted resulting in fast encryption and decryption speeds. The proposed encryption scheme consists of four stages of encryption. The first two stages involve permutation and the next two stages involve confusion. The proposed scheme is specifically designed to protect medical images and to facilitate fast encryption, transmission, and decryption of the input images. The procedural flowchart of the proposed scheme mentioning all components is depicted in Fig. 1.

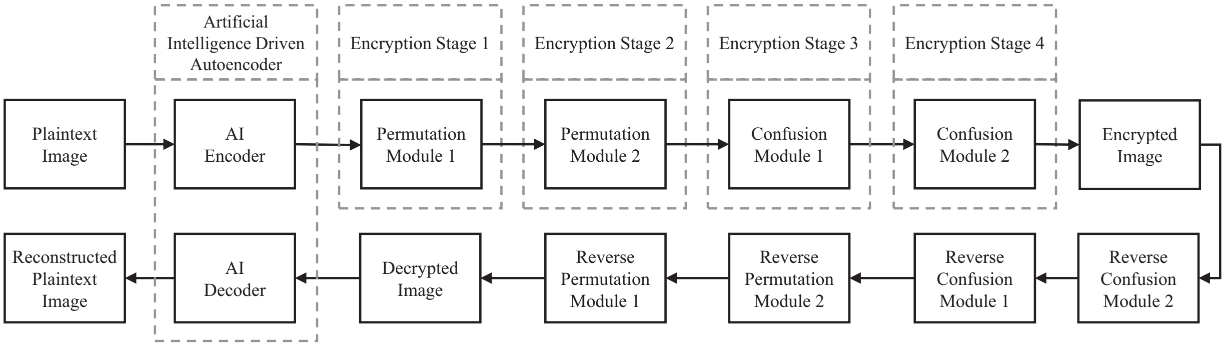


Figure 1: Flowchart of the proposed artificial intelligence-driven encryption scheme

The main contributions of this paper are:

1. A novel AI-based convolutional autoencoder is proposed. This autoencoder encodes the HD medical images into lower dimensional space to facilitate fast encryption and transmission of medical images over various communication channels.

2. A novel four-stage medical image encryption is proposed. The first two stages involve two permutation modules that ensure the disruption of the correlation within the medical images. The last two stages involve two modules of confusion which makes the proposed scheme resistant to statistical and known-plaintext attacks.

The rest of the paper is organized as follows. A detailed literature review is presented in Section 2 which summarises the latest work related to this paper and highlights the research gap. Section 3 presents the design of the autoencoder. Section 4 entails the proposed medical image encryption scheme. The results are presented in Section 5, which is followed by a clear and concise conclusion in Section 6.

2 Related Work

The necessity for secure transmission and storage of medical images in telemedicine and related fields has resulted in the development of various cryptographic schemes. These systems aim to safeguard sensitive patient information from unauthorized access and misuse. For instance, a novel encryption system for digital medical images using DNA cryptography and dual hyper-chaotic maps is proposed in [11]. This scheme utilizes DNA encoding for the encryption stage and performs selective pixel encryption. This technique ensures fast encryption/decryption phases and results in efficient protection of medical images. In a similar attempt to protect medical images, the authors in [12] also follow the approach of elective encryption. They have utilized chaotic maps in conjunction with DNA encryption to protect medical images and the results portray secure encryption. Moreover, different chaotic maps have been utilized in [13]. The authors have utilized Chen's chaotic system and Henon map and used it with Brownian motion. The authors claim that it is a lightweight encryption system and evaluation shows commendable statistical security results. Furthermore, hybrid chaotic maps have also been utilized to protect medical images, for example, the authors in [14] use a two-dimensional Logistic-Sine-Coupling map with the Arnlod's cap map. Utilization of these two maps together results in a secure encryption scheme with close-to-ideal results. Moreover, medical images have also been encrypted in the frequency domain, like the authors in [15] propose a frequency domain encryption scheme and utilize wavelet transform in conjunction with DNA encryption. They also proposed a key generation module using 3D Lorenz map. The authors performed several security evaluations and the scheme performed very well.

Speaking of the application of autoencoders in image encryption schemes, several encryption schemes have utilized autoencoders, such as the authors in [9] present a convolutional autoencoder-based encryption scheme that utilizes DNA encryption to encrypt images. The autoencoder reconstructs images exceptionally well and the evaluation of the encryption scheme also showcases close to ideal results. Besides in another paper [16], a deep autoencoder has been used to maintain uniform distribution and to ensure randomness created by the encryption scheme. Furthermore, autoencoders have also been utilized in conjunction with chaotic maps in [17]. The logistic map is used for image scrambling and the autoencoder is used for dimensionality reduction. In addition, autoencoders have also been used as image-scrambling adversaries in [18]. The proposed scheme utilizes asymmetric encryption. The autoencoder is a Cycle-Consistent Generative Adversarial Network (CycleGAN) network and results in a secure encryption system. Besides, the autoencoders have also been used in an encryption scheme proposed in [19]. The autoencoder is used for semantic feature extraction, which follows a carefully designed encryption scheme. This scheme also utilizes an attention mechanism which is used to optimise the compression mechanism. Results indicate a high level of security and effective image compression. While most of the papers focus on dimensionality reduction, the architecture of the autoencoders is not lightweight. This paper focuses on proposing a new architecture that is lightweight and offers exact reconstruction of input images.

3 Design of the Proposed AI Autoencoder

In this paper, an autoencoder is proposed for the effective compression of high-resolution medical images before encryption to facilitate fast encryption-decryption stages. The architecture of the proposed autoencoder is given in Fig. 2 and Table 1. The autoencoder has been trained on two public datasets, i.e., the 'COVID-19 Pakistani Patients X-ray Image Dataset' [20,21] and the 'Brain MRI Images for Brain Tumor Detection' [22], both are publicly available on Kaggle. The training performance is given in the accuracy and loss curves in Fig. 3. The curves show that the autoencoder exhibits almost 99% accuracy. The following subsections entail the modeling of the proposed autoencoder.

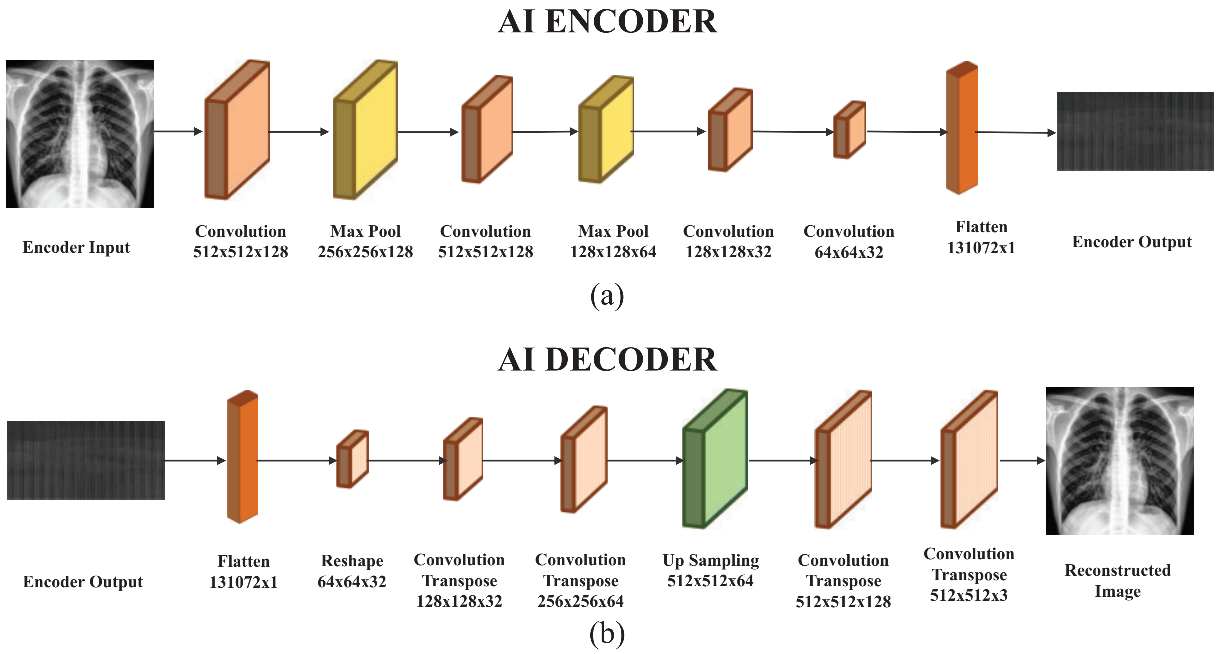


Figure 2: Architecture of the proposed AI autoencoder; (a) AI Encoder. (b) AI Decoder

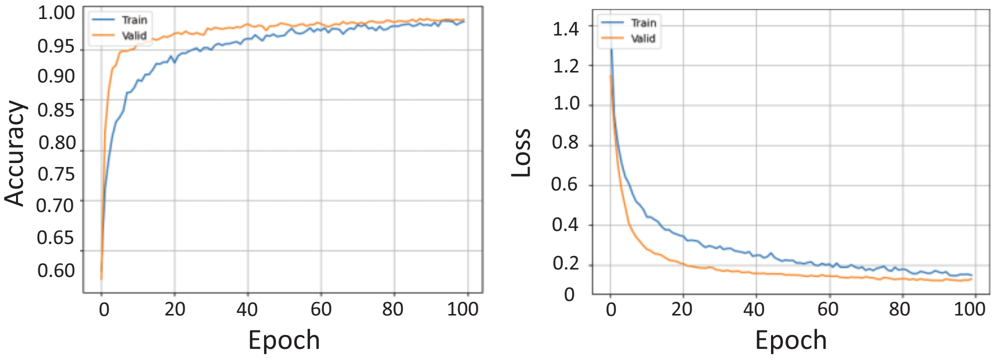
Table 1: Architecture of the autoencoder

Neural network layers	Feature map	Size	Kernel	Stride	Activation function
Input layer	Input image	–	512 × 512 × 3	–	–
1: Convolution	128	512 × 512 × 128	3 × 3	1	ReLU
2: Max pool	128	256 × 256 × 128	2 × 2	2	–

(Continued)

Table 1 (continued)

Neural network layers	Feature map	Size	Kernel	Stride	Activation function
3: Convolution	64	256 × 256 × 64	3 × 3	1	ReLU
4: Max pool	64	128 × 128 × 64	2 × 2	2	–
5: Convolution	32	128 × 128 × 32	3 × 3	1	ReLU
6: Convolution	32	64 × 64 × 32	3 × 3	2	ReLU
7: Flatten	–	131072 × 1	–	–	–
8: Dense (Encoder-Out)	–	256	–	–	–
9: Dense (Decoder-In)	–	256	–	–	–
10: Reshape	–	64 × 64 × 32	–	–	–
11: Convolution transpose	32	128 × 128 × 32	3 × 3	2	ReLU
12: Convolution transpose	64	256 × 256 × 64	3 × 3	2	ReLU
13: UpSampling2D	64	512 × 512 × 64	2 × 2	1	–
14: Convolution transpose	128	512 × 512 × 128	3 × 3	1	ReLU
15: Convolution transpose	3	512 × 512 × 3	3 × 3	1	ReLU
Output	Reconstructed image	–	512 × 512 × 3	–	–



(a) (b)

Figure 3: Training performance of the AI autoencoder; (a) Accuracy curves. (b) Loss curves

3.1 Mathematical Modelling of the Autoencoder Architecture

A comprehensive mathematical modeling of the proposed autoencoder including the encoder and decoder is given as follows:

Design of the Encoder

1. The autoencoder begins with an input layer receiving an RGB image of dimensions $512 \times 512 \times 3$, denoted by the tensor $X \in \mathbb{R}^{512 \times 512 \times 3}$. This foundational layer facilitates the extraction of features by subsequent layers.

$$X \in \mathbb{R}^{512 \times 512 \times 3} \tag{1}$$

2. The first convolutional layer employs 128 filters, each of size 3×3 and stride 1. This layer applies these filters to generate feature maps, using the Rectified Linear Unit (ReLU) activation function to introduce non-linearity. The mathematical operation for this convolutional layer is captured by:

$$Z_1[i,j,k] = \text{ReLU}\left(\sum_{m=-1}^{1}\sum_{n=-1}^{1}\sum_{c=1}^{3} X[i+m, j+n, c] \cdot W_1[m,n,c,k] + b_1[k]\right) \tag{2}$$

The parameter count for this layer is computed considering the number of filters, the dimensions of each filter, and the number of input channels:

$$\begin{aligned}\text{Parameters} &= \text{Number of Filters} \times (\text{Filter Height} \times \text{Filter Width} \times \text{Input Channels} + 1)\\ &= 128 \times (3 \times 3 \times 3 + 1) = 3,488\end{aligned} \tag{3}$$

3. A max pooling layer follows, with a 2×2 kernel and stride of 2, reducing the spatial dimensions to 256×256 and minimizing the potential for overfitting by downsampling feature maps:

$$Z_2[i,\ j,\ k] = \max_{0 \le m,\ n < 2} Z_1[2i+m,\ 2j+n,\ k] \tag{4}$$

4. Further, the third convolutional layer continues the feature extraction process. The third layer uses 64 filters with 3×3 kernels and a stride of 1, enhancing the detail of the features:

$$Z_3[i,\ j,\ k] = \text{ReLU}\left(\sum_{m=-1}^{1}\sum_{n=-1}^{1}\sum_{c=1}^{128} Z_2[i+m,\ j+n,\ c] \cdot W_3[m,\ n,\ c,\ k] + b_3[k]\right) \tag{5}$$

Including the effects of stride on the output dimensions, this layer helps further reduce dimensionality efficiently, which can be generally calculated as:

$$\text{Output Dimension} = \left\lfloor \frac{\text{Input Dimension} - \text{Filter Size} + 2 \times \text{Padding}}{\text{Stride}} \right\rfloor + 1 \tag{6}$$

5. Another max pooling layer reduces dimensions to 128×128, further compressing the data:

$$Z_4[i,\ j,\ k] = \max_{0 \le m,\ n < 2} Z_3[2i+m,\ 2j+n,\ k] \tag{7}$$

6. Subsequent layers, including further convolutional stages and dense layers, process and transform the compressed feature maps into a more abstract representation, suitable for the encoding process:

$$Z_5[i, j, k] = \text{ReLU}\left(\sum_{m=-1}^{1}\sum_{n=-1}^{1}\sum_{c=1}^{64} Z_4[i+m, j+n, c] \cdot W_5[m, n, c, k] + b_5[k]\right) \tag{8}$$

$$Z_6[i, j, k] = \text{ReLU}\left(\sum_{m=-1}^{1}\sum_{n=-1}^{1}\sum_{c=1}^{32} Z_5[2i+m, 2j+n, c] \cdot W_6[m, n, c, k] + b_6[k]\right) \tag{9}$$

7. The encoded feature map is flattened and processed through dense layers, transforming the learned features into a compressed form that captures the essence of the input data:

$$Z_7 = \text{vec}(Z_6) \tag{10}$$

$$Z_8 = \text{ReLU}(W_8 Z_7 + b_8) \tag{11}$$

Design of the Decoder

Decoder layers, including convolutional transpose operations, reconstruct the original image from the compressed representation:

$$Z_9 = \text{ReLU}(W_9 Z_8 + b_9) \tag{12}$$

$$Z_{10} = \text{reshape}(Z_9, (64 \times 64 \times 32)) \tag{13}$$

$$Z_{11}[i, j, k] = \text{ReLU}\left(\sum_{m=-1}^{1}\sum_{n=-1}^{1} Z_{10}[i/2+m, j/2+n, c] \cdot W_{11}[m, n, c, k] + b_{11}[k]\right) \tag{14}$$

$$Z_{12}[i, j, k] = \text{ReLU}\left(\sum_{m=-1}^{1}\sum_{n=-1}^{1} Z_{11}[i/2+m, j/2+n, c] \cdot W_{12}[m, n, c, k] + b_{12}[k]\right) \tag{15}$$

$$Z_{13}[i, j, k] = Z_{12}[i/2, j/2, k] \tag{16}$$

Finally, the output layer consists of a convolutional transpose layer that uses three filters to reconstruct the RGB channels of the final output image, returning it to its original dimensions and appearance:

$$Y[i, j, k] = \text{ReLU}\left(\sum_{m=-1}^{1}\sum_{n=-1}^{1} Z_{14}[i+m, j+n, c] \cdot W_{15}[m, n, c, k] + b_{15}[k]\right) \tag{17}$$

4 The Proposed Medical Encryption Scheme

The proposed AI-driven encryption scheme comprises two important constituents, i.e., the novel AI-based autoencoder and the proposed medical image encryption scheme. The design of the autoencoder was explained in the previous section. The following subsections entail the design and implementation of the encryption scheme. The proposed encryption scheme is a four-stage encryption scheme. There are two stages of permutation and two stages of confusion. This four-stage encryption scheme ensures effective encryption. The detailed flowchart of the proposed medical image encryption scheme is given in Fig. 4.

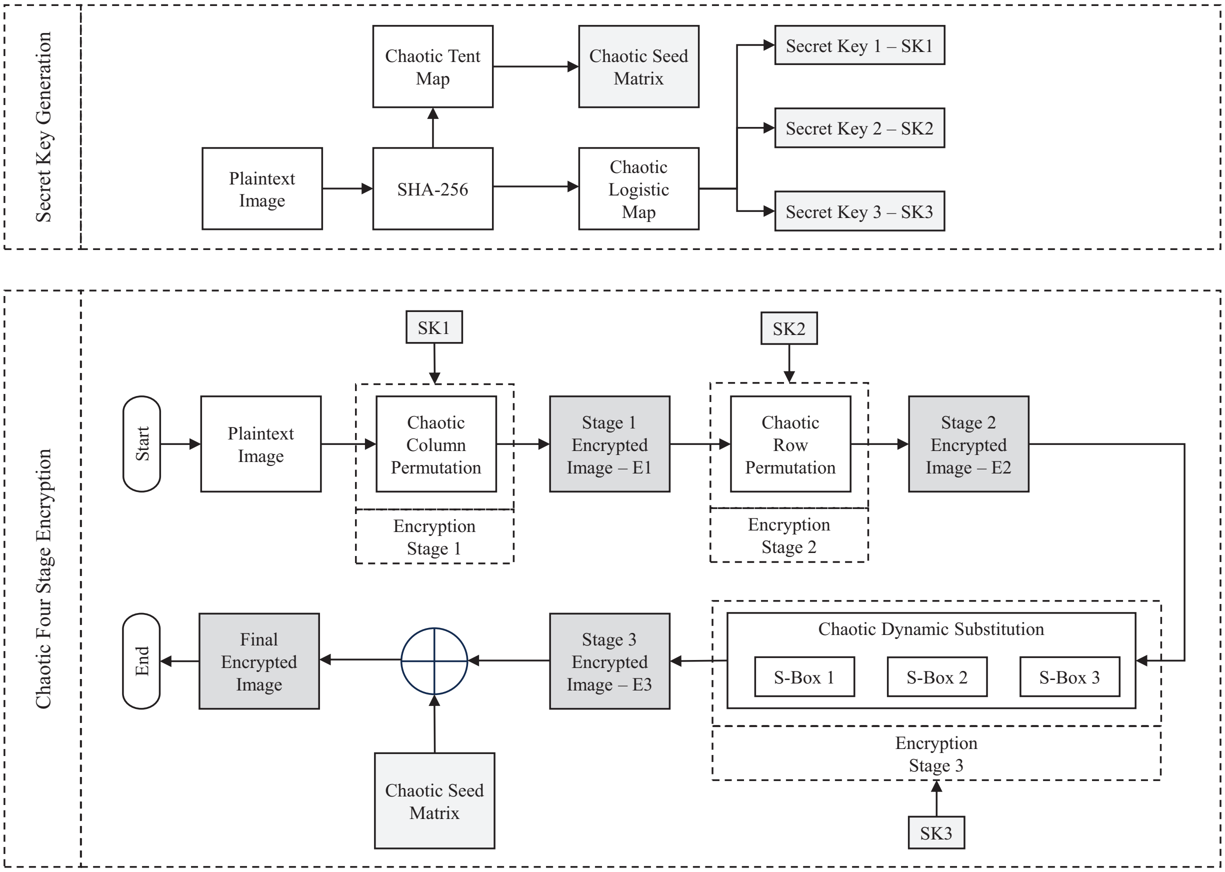


Figure 4: Flowchart of the proposed medical image encryption scheme

1. **Hashing the Input:** The first step is to read the encoded image and to apply the 256-bit Secure Hash Algorithm (SHA-256) hash to generate a unique hash. This unique hash is used to create three secret keys and a seed matrix. The secret keys control the permutation and the confusion process and the seed matrix is used in the fourth stage of the encryption. SHA-256 generates a 256-bit hash value from any input data. The algorithm processes a block size of 512 bits and outputs a hash of 256 bits. The input message undergoes padding to make its length a multiple of 512 bits. The steps include:
   (a) Append a '1' bit to the message.
   (b) Append '0' bits until the message length is 448 modulo 512.
   (c) Append the original message length as a 64-bit number.

   The SHA-256 function uses several logical functions and bitwise operations:
   - $Ch(x, y, z) = (x \wedge y) \oplus (\neg x \wedge z)$
   - $Maj(x, y, z) = (x \wedge y) \oplus (x \wedge z) \oplus (y \wedge z)$
   - $\Sigma_0(x) = ROTR^2(x) \oplus ROTR^{13}(x) \oplus ROTR^{22}(x)$
   - $\Sigma_1(x) = ROTR^6(x) \oplus ROTR^{11}(x) \oplus ROTR^{25}(x)$
   - $\sigma_0(x) = ROTR^7(x) \oplus ROTR^{18}(x) \oplus SHR^3(x)$
   - $\sigma_1(x) = ROTR^{17}(x) \oplus ROTR^{19}(x) \oplus SHR^{10}(x)$

The following steps are followed while generating the hash:

(a) Initialize the hash values $h_0,\ h_1, \ldots, h_7$.

(b) Expand the 512-bit block into a 64-word schedule using the σ functions.

(c) For each round, update the working variables and mix them into the intermediate hash values using the Σ and Ch, Maj functions along with round constants.

(d) After processing all blocks, concatenate the intermediate hash values to form the final hash.

2. **Initialization of the Chaotic Map:** In this step, the logistic chaotic map is initialized by setting its control parameter and initial conditions. The logistic map equation used is:

   $x_{n+1} = r \cdot x_n \cdot (1 - x_n);$

   This map takes input from the unique hash and generates the three secret keys and the seed matrix.

3. **Encryption Stage 1: Chaotic Column Shuffling:** The first secret key derived from the logistic map sequence is used to shuffle the columns of the image. This shuffling is based on a permutation generated by sorting the outputs of several iterations of the map. The following steps are involved in the encryption stage 1:

   - Selecting an initial value x_0, obtained from the hash function:

     $x_0 = \text{hash value}$

   - Iterating the logistic map for N iterations to reach the chaotic regime:

     $x_{i+1} = r \cdot x_i \cdot (1 - x_i) \quad \text{for } i = 0,\ 1,\ 2, \ldots, N - 1$

   - Recording the values after discarding initial transients to form sequence S:

     $S = [x_k,\ x_{k+1}, \ldots, x_{k+M-1}]$

     where k is the number of initial values discarded, and M is the number of columns to be shuffled.

   - Converting the chaotic sequence to column indices for a matrix with m columns:

     $\text{index}_i = \lfloor x_i \cdot m \rfloor \quad \text{for } i = k,\ k + 1, \ldots, k + M - 1$

   - In this step, for a matrix A of dimensions $n \times m$ where n is the number of rows and m is the number of columns, the columns are shuffled as follows:

     $A'[:, i] = A[:, \text{index}_i] \quad \text{for each column } i$

     This reorders the columns based on the indices generated from the logistic map sequence. Following is the example of the encryption stage 1 on a matrix A with 5 columns ($m = 5$), with initial conditions of the logistic maps as $r = 3.9$, and $x_0 = 0.5$. Then:

     $x_1 = 3.9 \cdot 0.5 \cdot (1 - 0.5) = 0.975$

     $x_2 = 3.9 \cdot 0.975 \cdot (1 - 0.975) = 0.09350625$

     $\vdots$

The values from the chaotic indices are:

$\text{index}_1 = \lfloor 0.975 \cdot 5 \rfloor = 4$

$\text{index}_2 = \lfloor 0.09350625 \cdot 5 \rfloor = 0$

$\vdots$

The columns are shuffled accordingly:

$A'[:, 0] = A[:, 4]$

$A'[:, 1] = A[:, 0]$

$\vdots$

This process repeats for all columns of the image.

4. **Encryption Stage 2: Row Permutation:** In this stage, a permutation key from the logistic map, i.e., the secret key 2 is used to permute the rows of the column-permuted image in a similar way done in stage 1.

   $Q[:, j] = P[\tau(j), :]$

   where τ is a permutation of the rows based on the chaotic sequence.

5. **Encryption Stage 3: Dynamic S-Box Substitution:** In this stage, a dynamic S-box substitution is applied, and the secret key 3, which is also generated from the chaotic map, is utilized to select any one of the 3 S-boxes. The values from the selected s-box replace pixels in the row and column-shifted image. This substitution process ensures confusion the image.

   $R[i, j] = S_k[P[i, j]]$

   where S_k is selected from a set of predefined S-boxes based on the key. The substitution process is performed as follows:

   - Producing a chaotic sequence to select S-boxes:

     $x_{i+1} = r \cdot x_i \cdot (1 - x_i)$

   - Maping each x_i to one of three S-boxes $S_1,\ S_2,\ S_3$.
   - Converting each pixel p to 8-bit binary, then splitting into Most Significant Bits (MSB) and Least Significant Bits (LSB), converting back to decimal, and calculating the S-box index:

     $\text{MSB}_{\text{dec}} = \text{decimal value of MSB}, \quad \text{LSB}_{\text{dec}} = \text{decimal value of LSB}$

     $\text{S-box index} = \text{MSB}_{\text{dec}} \times \text{LSB}_{\text{dec}}$

   - Substituting by using the selected S-box:

     $p' = S_{\text{index}_i}[\text{S-box index}]$

6. **Encryption Stage 4: Seed Matrix Addition:** The final encryption layer involves bitwise Exclusive OR (XOR) with a seed matrix derived from the logistic map to enhance security:

   $S[i, j] = R[i, j] \oplus M[i, j]$

where M is the seed matrix, adding another layer of confusion and diffusion.

5 Results and Analysis

The proposed encryption scheme has been evaluated for various statistical security parameters to validate its security characteristics. Histogram analysis, correlation analysis, entropy analysis, correlation analysis, and other statistical parameters, such as energy, contrast, and homogeneity have been evaluated. The visual encryption results of four stages are given in Fig. 5.

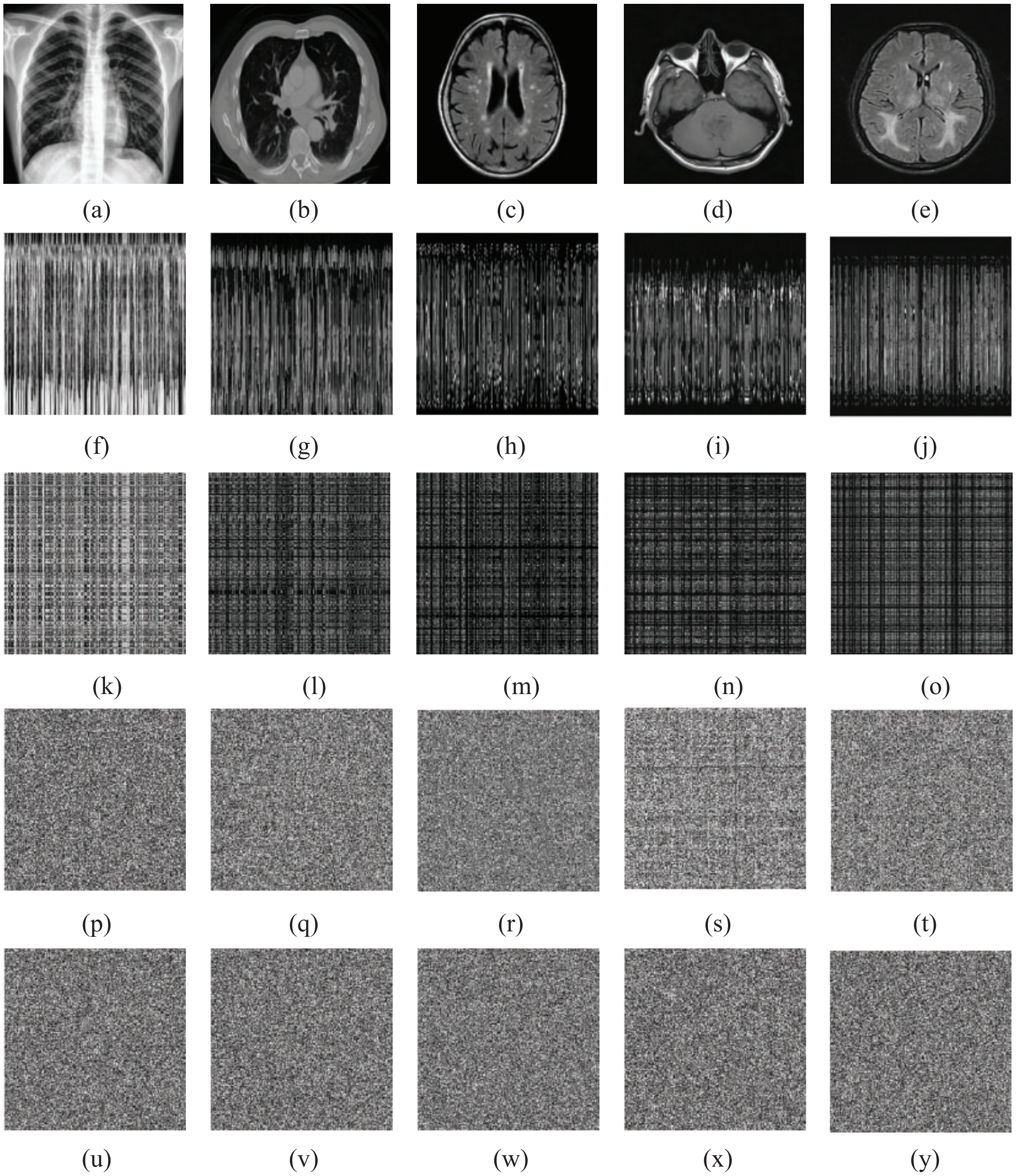


Figure 5: Encryption results of four-stage encryption scheme. (a) to (e) Plaintext images. (f) to (j) Stage 1. (k) to (o) Stage 2. (p) to (t) Stage 3. (u) to (y) Stage 4

5.1 Histogram Analysis

The histogram of an image represents the frequency of occurrence of intensity levels within the image. This information is crucial for attackers as it can be used to extract concealed information from an encrypted image. Therefore, an ideal secure encryption scheme should result in a flat or uniformly distributed histogram, meaning that all intensity levels occur an equal number of times. As shown in the histogram analysis in Fig. 6, the histograms of the ciphertext images are uniformly distributed, depicting that all pixel intensities from 0 to 255 occur an equal number of times. Hence, the original information regarding the frequency of occurrence cannot be retrieved. These histograms demonstrate the effectiveness of the proposed encryption scheme.

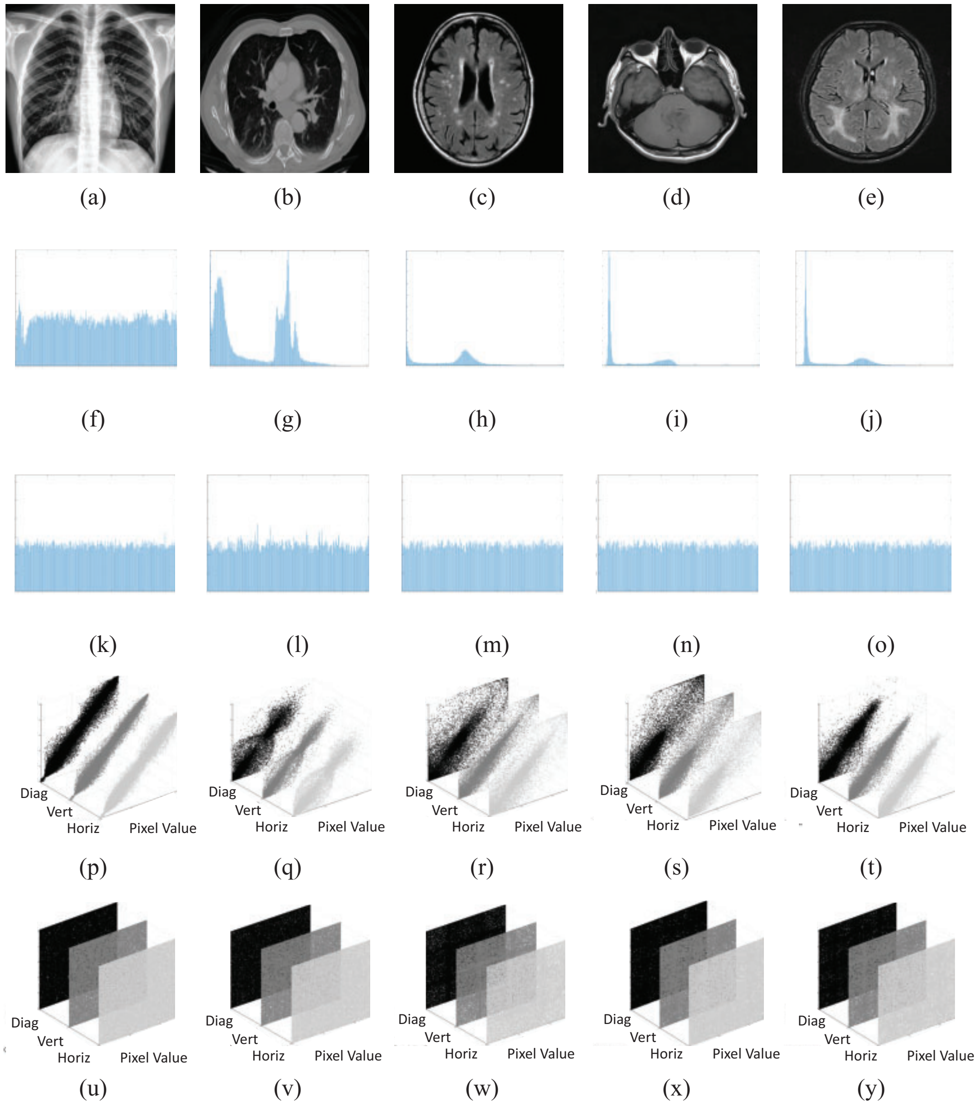


Figure 6: Histogram and correlation analysis. (a) to (e) Plaintext images. (f) to (j) Histogram of plaintext images. (k) to (o) Histogram of ciphertext images. (p) to (t) Correlation of plaintext images. (u) to (y) Correlation analysis of ciphertext images

5.2 Correlation Analysis

Correlation within an image indicates the patterns and shapes present in the image. Areas with defined patterns and shapes exhibit high correlation, whereas areas where pixels are not correlated with each other display low correlation values. An ideally encrypted image should correlate close to zero, indicating that no pixel is related to its neighbor. The correlation plots of an ideally encrypted image should be widely scattered, demonstrating the disrupted correlation throughout the image. As illustrated in Fig. 6, the correlation plots are widely scattered depicting that neighboring pixels in the cipher image are not correlated, validating the effectiveness of the proposed encryption scheme. Moreover, it can be seen in Table 2 that the correlation within the image is ideally close to zero. The table presents the correlation of plaintext and ciphertext images individually by calculating the Gray Level Co-occurrence Matrix (GLCM) and the correlation between the plaintext and ciphertext images as well.

Table 2: Correlation analysis of the proposed encryption scheme

Parameter	Images	Image 1	Image 2	Image 3	Image 4	Image 5
Correlation	Plaintext (GLCM)	0.970	0.955	0.898	0.935	0.945
	Ciphertext (GLCM)	0.001	−0.001	0.001	0.0004	0.002
	Between plaintext and ciphertext	0.007	0.001	0.003	0.003	0.001

5.3 Entropy Analysis

Entropy, or information entropy, measures the amount of randomness in an image. The greater the randomness, the more secure the encrypted image is. For a grayscale image, the ideally encrypted image should have an entropy of 8, which indicates that there is maximum randomness in the image, making it difficult to retrieve any information. As shown in Table 3, the entropy of the ciphertext images is ideally close to 8.

Table 3: Entropy analysis of the proposed encryption scheme

Parameter	Images	Image 1	Image 2	Image 3	Image 4	Image 5
Entropy	Plaintext	7.980	6.560	4.745	5.757	6.085
	Ciphertext	7.998	7.994	7.994	7.982	7.977

5.4 Contrast Analysis

Contrast in an image indicates how rapidly the intensity changes between two pixels or within a certain area. In regions where there are distinct shapes, the contrast is usually low because the pixels are related to each other to form a shape. Conversely, areas where colors change abruptly exhibit high contrast. An ideally encrypted image should have high contrast, indicating that no two pixels are related and the intensity of every neighboring pixel is completely different from each other. As shown in Table 4, the contrast of the ciphertext images is very high, validating the effectiveness of the proposed encryption scheme.

Table 4: Contrast analysis of the proposed encryption scheme

Parameter	Images	Image 1	Image 2	Image 3	Image 4	Image 5
Contrast	Plaintext	0.300	0.237	0.467	0.346	0.254
	Ciphertext	10.485	10.194	9.239	10.544	10.116

5.5 Energy Analysis

The energy of an image refers to the uniformity and repetition of pixels within it. High energy indicates the presence of patterns and shapes, while low energy values suggest that the pixels are unrelated and there are no distinct shapes in an image. Therefore, ideally, the energy of a well-encrypted image should be low. This can be validated through Table 5, which shows that the energy values of the encrypted images are quite low.

Table 5: Energy analysis of the proposed encryption scheme

Parameter	Images	Image 1	Image 2	Image 3	Image 4	Image 5
Energy	Plaintext	0.074	0.285	0.342	0.345	0.295
	Ciphertext	0.015	0.015	0.016	0.015	0.015

5.6 Homogeneity Analysis

Homogeneity measures the consistency and smoothness of patterns across an image. High homogeneity indicates smooth areas with distinct shapes, whereas low homogeneity corresponds to areas with no consistent relationship among pixels. An ideal encrypted image should have low homogeneity, indicating that the pixel values are less correlated. This can be validated by the results in Table 6, which show that all homogeneity values are very low, depicting secure encryption.

Table 6: Homogeneity analysis of the proposed encryption scheme

Parameter	Images	Image 1	Image 2	Image 3	Image 4	Image 5
Homogeneity	Plaintext	0.869	0.922	0.897	0.915	0.909
	Ciphertext	0.389	0.390	0.398	0.388	0.392

6 Conclusion

This paper introduced an innovative AI-driven encryption scheme specifically tailored for securing medical images. The process starts by utilizing an AI-based autoencoder that compresses high-resolution images into lower-resolution versions while preserving essential diagnostic information. These compressed images then undergo a sophisticated four-stage encryption process, involving permutation and confusion techniques, to enhance security. The permutation stages disrupt the image structure, effectively safeguarding against statistical attacks, while the confusion stages obscure the pixel values, making unauthorized decryption challenging. The effectiveness of this encryption strategy was rigorously validated through extensive security and performance evaluations. Results from these tests demonstrated that the scheme effectively maintained the confidentiality and integrity of the medical images during storage and transmission. This robust encryption approach thus ensures that sensitive medical data is protected against various cyber threats, making it highly suitable for real-world application in healthcare settings where data security is paramount. Moreover, a possible limitation of the scheme could be the scalability issue. As the volume of data increases, especially in large healthcare systems generating vast amounts of medical images, the training process could take more time.

Acknowledgement: None.

Funding Statement: The author received no specific funding for this research.

Availability of Data and Materials: All data generated or analyzed during this study are included in this published article.

Ethics Approval: Not applicable.

Conflicts of Interest: The author declares that they have no conflicts of interest to report regarding the present study.

References

1. Ahmed ST, Hammood DA, Chisab RF, Al-Naji A, Chahl J. Medical image encryption: a comprehensive review. Computers. 2023;12(8):160. doi: 10.3390/computers12080160.
2. Lai Q, Hu G, Erkan U, Toktas A. High-efficiency medical image encryption method based on 2D logistic-Gaussian hyperchaotic map. Appl Math Comput. 2023;442:127738.
3. Zhou NR, Hu LL, Huang ZW, Wang MM, Luo GS. Novel multiple color images encryption and decryption scheme based on a bit-level extension algorithm. Expert Syst Appl. 2024;238:122052. doi: 10.1016/j.eswa.2023.122052.
4. Abbasi SF, Ahmad J, Khan JS, Khan MA, Sheikh SA. Visual meaningful encryption scheme using intertwinning logistic map. In: Intelligent Computing: proceedings of the 2018 Computing Conference, 2019; Cham: Springer; vol. 2, p. 764–73.
5. Khan MS, Ahmad J, Ali H, Pitropakis N, Al-Dubai A, Ghaleb B, et al. SRSS: a new chaos-based single-round single S-box image encryption scheme for highly auto-correlated data. In: 2023 International Conference on Engineering and Emerging Technologies (ICEET), 2023; Istanbul, Turkey; p. 1–6.
6. Khan MS, Ahmad J, Al-Dubai A, Ghaleb B, Pitropakis N, Buchanan WJ. RNA-TransCrypt: Image encryption using chaotic RNA encoding, novel transformative substitution, and tailored cryptographic operations. 2024. doi:10.48550/arXiv.2401.04707.
7. Khan MS, Ahmad J, Al-Dubai A, Jaroucheh Z, Pitropakis N, Buchanan WJ, et al. PermutEx: feature-extraction-based permutation—a new diffusion scheme for image encryption algorithms. In: 2023 IEEE 28th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 2023; Edinburgh, UK; p. 188–93.
8. Ali H, Khan MS, Driss M, Ahmad J, Buchanan WJ, Pitropakis N. CellSecure: securing image data in industrial internet-of-things via cellular automata and chaos-based encryption. In: 2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall), 2023; Hong Kong, China: IEEE; p. 1–6.
9. Ahmed F, Rehman MU, Ahmad J, Khan MS, Boulila W, Srivastava G, et al. A DNA based colour image encryption scheme using a convolutional autoencoder. ACM T Multim Comput. 2023;19(3s):1–21. doi:10.1145/3570165.

10. Alawida M, Teh JS, Alshoura W. A new image encryption algorithm based on DNA state machine for UAV data encryption. Drones. 2023;7(1):38. doi:10.3390/drones7010038.
11. Akkasaligar PT, Biradar S. Selective medical image encryption using DNA cryptography. Inform Secur J: A Glob Perspect. 2020;29(2):91–101. doi:10.1080/19393555.2020.1718248.
12. Kamal ST, Hosny KM, Elgindy TM, Darwish MM, Fouda MM. A new image encryption algorithm for grey and color medical images. IEEE Access. 2021;9:37855–65. doi:10.1109/ACCESS.2021.3063237.
13. Masood F, Driss M, Boulila W, Ahmad J, Rehman SU, Jan SU, et al. A lightweight chaos-based medical image encryption scheme using random shuffling and XOR operations. Wirel Pers Commun. 2022;127(2):1405–32. doi:10.1007/s11277-021-08584-z.
14. Jain K, Aji A, Krishnan P. Medical image encryption scheme using multiple chaotic maps. Pattern Recognit Lett. 2021;152:356–64. doi:10.1016/j.patrec.2021.10.033.
15. Banu SA, Amirtharajan R. A robust medical image encryption in dual domain: chaos-DNA-IWT combined approach. Med Biol Eng Comput. 2020;58:1445–58. doi:10.1007/s11517-020-02178-w.
16. Sang Y, Sang J, Alam MS. Image encryption based on logistic chaotic systems and deep autoencoder. Pattern Recognit Lett. 2022;153:59–66. doi:10.1016/j.patrec.2021.11.025.
17. Ameen Suhail K, Sankar S. Image compression and encryption combining autoencoder and chaotic logistic map. Iran J Sci Technol, Trans A: Sci. 2020;44(4):1091–100. doi:10.1007/s40995-020-00905-4.
18. Bao Z, Xue R, Jin Y. Image scrambling adversarial autoencoder based on the asymmetric encryption. Multimed Tools Appl. 2021;80(18):28265–301. doi:10.1007/s11042-021-11043-3.
19. Wang B, Lo KT. Autoencoder-based joint image compression and encryption. J Inf Secur Appl. 2024;80:103680. doi:10.1016/j.jisa.2023.103680.
20. Khan MS. COVID-19 Pakistani patients X-ray image dataset. 2021. Available from: https://www.kaggle.com/datasets/muhammadshahbazkhan/covid19-pakistani-patients-xray-image-dataset. [Accessed 2024].
21. Umair M, Khan MS, Ahmed F, Baothman F, Alqahtani F, Alian M, et al. Detection of COVID-19 using transfer learning and grad-CAM visualization on indigenously collected X-ray dataset. Sensors. 2021;21(17):5813. doi:10.3390/s21175813.
22. Chakrabarty N. Brain MRI images for brain tumor detection. 2019. Available from: https://www.kaggle.com/datasets/navoneel/brain-mri-images-for-brain-tumor-detection. [Accessed 2024].