IRJCS: International Research Journal of Computer Science
Volume 11, Issue 11, December 2024
ISSN: 2393-9842
https://www.irjcs.com/archives

Machine Learning and Secure Image Transmission for Disease Forecasting

Gaurav Sharma
Department of Information Technology,
Greater Noida Institute of Technology (Engg. Institute)
Greater Noida, India
gs9105@outlook.com
Mandeep Chaudhary
Department of Information Technology,
Greater Noida Institute of Technology (Engg. Institute)
Greater Noida, India
mandeepchaudhary221@gmail.com
Hariom Kumar
Department of Information Technology,
Greater Noida Institute of Technology (Engg. Institute)
Greater Noida, India
Dr.Ajay Kumar Sahu
Professor, Department of Information Technology,
Greater Noida Institute of Technology (Engg. Institute), Greater Noida, India
ajaysahu.it@gniot.net.in

Publication History
Manuscript Reference No: IRJCS/RS/Vol.11/Issue11/DCCS10083 | Research Article | Open Access | Double-Blind Peer Reviewed **Article ID:** IRJCS/RS/Vol.11/Issue11/DCCS10083
Received:28,November2024,Revised:04,December2024Accepted:10December2024PublishedOnline: 16, December 2024
http://www.irjcs.com/volumes/Vol11/iss-11/04.DCCS10083.pdf
Article Citation: Gaurav, Mandeep, Hariom, Dr.Ajay (2024). Machine Learning and Secure Image Transmission for Disease Forecasting. IRJCS: International Research Journal of Computer Science, Volume 11, Issue 10 of 2024 pages 652-658 **doi:>** https://doi.org/10.26562/irjcs.2024.v1111.04
BibTeX Gaurav@2024Machine

I. INTRODUCTION

In today's era of information technology, the rapid expansion of the Internet, especially in the realm of electronic healthcare, has become both feasible and widespread. Internet-enabled electronic healthcare systems allow patients to consult with specialist doctors remotely for diagnoses. Medical images are frequently stored, processed, and transmitted online, and during this process, sensitive data may be involved, requiring strict privacy measures. Given the importance of protecting patient information, encrypting this data is crucial to ensure that only authorized individuals can access it. Encrypting medical images presents unique challenges, particularly because of the need for fast and accurate data extraction. Traditional encryption methods may not always be suitable for securing large medical images effectively. Therefore, it's essential to safeguard the algorithms used for processing these images from potential attacks. One commonly used encryption tool is a random number generator, which produces sequences of numbers that are not predictable. The more unpredictable the number generation, the stronger the encryption, which makes it more effective. Chaos systems are often employed to generate pseudo-random numbers, providing a robust encryption key.

Certain techniques are particularly effective in generating initial random numbers with significant power, sensitivity to initial conditions, long frequencies, and the ability to produce large encryption keys. This paper seeks to combine chaotic systems with key optimization to improve both security and speed. For an encryption algorithm to be effective, it must be secure and efficient in real-time applications. However, many current algorithms are unsuitable for real-time use due to long sequences and poor performance during online transmission. Additionally, these algorithms can be vulnerable to external attacks, especially when data is transmitted online. Advancements in device performance, particularly in processor speed, have led to improvements in algorithms, especially those related to random number generation. Optimizing the use of a specific processor involves choosing between two hardware options: ASIC (Application-Specific Integrated Circuit) and FPGA (Field-Programmable Gate Array). While ASICs are more expensive, FPGAs offer a promising solution. FPGAs are efficient because they allow designers to leverage programmable logic elements to develop intelligent algorithms. However, they can be costly in terms of design complexity. On the other hand, ASICs can deliver the required performance at a lower cost; though developing them involves significant effort. The key advantage of ASICs is their ability to execute tasks at high speeds, making them more suitable for hardware processing than relying on software solutions. In recent years, Deep Learning (DL) and Artificial Intelligence (AI) have made substantial progress, particularly in the field of medical imaging. AI techniques play a vital role in processing and diagnosing medical images, including tasks such as interpretation, merging, recording, and segmentation.

Many medical images are obtained through radiation-based imaging devices, and deep learning is used to analyze and extract valuable information from these images. This technology assists doctors in making accurate and timely diagnoses, enabling the early detection of diseases. AI helps in differentiating between images and identifying the root causes of medical conditions. Various AI techniques, including Support Vector Machines (SVM), Neural Networks (NN), K-Nearest Neighbor (KNN), Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) networks, and Extreme Learning Machines (ELM), have been applied to medical image analysis. Generative Adversarial Networks (GANs) are also used, though they require significant processing time. These algorithms analyze raw image data and classify it based on learned patterns. Deep learning enables systems to learn complex abstractions and identify key features from large image datasets, often sourced from standard databases, to enhance diagnostic accuracy.

While traditional diagnostic methods in medical imaging have remained reliable over time, deep learning has introduced a breakthrough in algorithm performance, allowing for higher accuracy and efficiency. Speed and precision have become crucial factors in medical image processing. Deep learning has demonstrated its effectiveness in a variety of fields, including speech recognition, text recognition, and computer-aided diagnosis. The aim of this study is to develop a method for processing medical images using deep learning, with an emphasis on optimizing the number of hidden layers in the neural network. The main contribution of this paper is to identify a chaotic system that remains undetectable when applied with the Deep Neural Network (DNN) algorithm, and to modify the DNN by focusing on the weights that have the greatest impact on the output, whether they correspond to intermediate or semi-final stages. The structure of this manuscript is organized as follows: Section 2 reviews existing research relevant to the topic and discusses the key studies in the field. Section 3 outlines the methodology adopted in this study, followed by the results presented in Section 4. The conclusions are drawn in Section 5.

II. RELATED WORK

Several previous studies have focused on the security of images in general [11], and medical images specifically [12]. The security of data and images heavily depends on the strength of the encryption algorithms used. Many different encryption methods have been proposed, many of which utilize robust random key generation techniques [13]. For example, a method based on cardiogram data for generating secure keys with reduced latency for encryption was proposed [14]. An enhanced technique for securing medical images using Fibonacci sequences to scatter and conceal image data was suggested by [15, 16]. Another approach employed the AES algorithm to generate random numbers using electrical impulse generators, thus improving image security [17]. A dynamic cipher system based on state estimation for cipher key generation was introduced by [18], which increases the randomness of the key over time, improving the security of medical images during transmission.

The use of chaotic systems has also been explored to enhance the security of medical images [20]. Research into linear systems has contributed to the development of encryption techniques by focusing on critical aspects such as sensitivity, predictability, pseudo-randomness, and certainty [21]. Additionally, AI-based methods for generating seeds have been proposed to further enhance the randomness and security of encryption keys. Tracking the distribution of data in medical images is often challenging, as it depends on complete randomness, which has proven effective through graphical and linear analysis [22]. A new method developed by recent researchers [23] uses a hybrid algorithm that demonstrates its effectiveness by generating chaos in the pixel distribution of images [24], making chaotic sequences difficult to trace unless the encryption key is used to recover the data [25]. Other researchers have focused on disrupting the relationships between pixels using specific equations, which track pixel effects and create identifiable paths through the encryption key. Artificial intelligence techniques have been integrated into the encryption of medical images to strengthen data security [26], with many researchers achieving significant results using AI-based algorithms such as Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), Support Vector Machines (SVM), Decision Trees, and K-Nearest Neighbors (KNN) [27]. Medical image diagnostics require innovative methods to identify abnormalities and monitor changes over time, and deep learning algorithms have greatly expanded the possibilities in this area. Medical images, primarily sourced from X-rays, CT scans, and MRIs, are processed using deep learning methods. Below are some diseases commonly diagnosed with deep learning techniques?

1. **Diabetic Retinopathy:** Diagnosing diabetic retinopathy (DR) manually can be difficult and time-consuming, as the disease often shows no early symptoms. Deep learning has proven accurate in analyzing retinal images. A Deep Convolutional Neural Network (DCNN) was employed to classify signals in eye images, achieving over 95% accuracy and sensitivity when trained on the EyePACS-1 dataset from 847 patients [28]. Other studies have utilized deep learning algorithms to detect bleeding, secretions, and aneurysms in retinal images [29].
2. **Histological and Microscopic Detection:** In histological analysis, changes in cells and surrounding tissues offer valuable insights for diagnosing diseases. Artificial intelligence algorithms, including DNNs, have been used to improve diagnostic accuracy. DNNs have been applied to detect cancer cells in colon biopsies [30, 31] and interstitial lung diseases using CNNs. CNNs have also been used in early breast cancer detection by classifying features from mammogram images [32].
3. **Gastrointestinal (GI) Diseases Detection:** Deep learning is critical in diagnosing gastrointestinal diseases using X-rays and MRIs. The DCNN method has been employed to detect hemorrhages in capsule endoscopy images [33]. Other studies have applied Fully Convolutional Networks (FCNs) and Long Short-Term Memory (LSTM) networks for feature extraction from large datasets [34].

4. Hybrid CNN-based methods have been used to detect digestive diseases from MRI images [35], and rapid feature extraction techniques using CNNs have been applied to detect inflammatory gastrointestinal diseases in WCT videos, with features classified by SVMs [36].
5. **Cardiac Imaging:** Deep learning has advanced cardiac imaging, especially in measuring calcium scores from MRI scans. SVMs have been used to classify extracted features and assist in diagnosis [38, 39]. Additionally, NN algorithms have classified heart conditions by evaluating data across multiple hidden layers for more precise results [40].
6. **Tumor Detection:** Tumors, whether benign or malignant, can be detected through medical imaging. A study used a method for diagnosing tumors from mammogram images in a dataset of 482 images, employing a median filter to reduce noise. Researchers frequently use SVM classifiers to distinguish benign from malignant tumors in mammograms. CNN algorithms analyze features extracted from radiographic images to measure clustering and classify the disease [41, 42].
7. **Alzheimer's and Parkinson's Diseases Detection:** These neurological disorders are linked to impaired motor function and the death of dopaminergic neurons. Alzheimer's disease can be diagnosed through clinical images and by observing shifts in fixed features in CT scans. A Deep Boltzmann Machine (DBM) was used to analyze distortions in 3D magnetic resonance images [43]. Another study used CNN to extract features from 3D MRI scans for Alzheimer's diagnosis, yielding satisfactory results with the CAD Dementia dataset for patients over 75 years old [44]. Furthermore, MRI brain images were used to detect Alzheimer's disease, achieving 98% accuracy when trained on a standard dataset [45].

III. PROPOSED METHOD

The methodology in this study is divided into two main parts. The first part focuses on improving data security in medical images through an enhanced encryption technique, while the second part involves classifying features extracted from the image using deep learning and the Deep Neural Network (DNN) algorithm. Encryption generally consists of two stages: confusion and diffusion. The main contribution of the proposed method lies in these two stages. For the confusion stage, pixel positions and sub-blocks are selected randomly, with the DNN algorithm assisting in this process. The DNN algorithm is also used to control the alteration of pixel values. Encryption is the process of obscuring image data to keep it secure and inaccessible to anyone without the decryption key. This method relies on irregularly scattering the image data, making it impossible to reconstruct without the encryption algorithm.

The proposed method is based on a chaotic system that functions in two stages: the Region of Interest (ROI) and the pixels within that region. Initially, a random key is generated, with chaotic systems playing a pivotal role in this process. The entropy of the image and its statistical randomness are carefully managed during this stage. The Henon Map method is used to generate the key because of its favorable characteristics, scalability, and compatibility with various techniques.

The key generation process is defined as follows:

xn+1=1−axn2+ynyn+1=bxn\begin{aligned} x_{n+1} &= 1 - a x_n^2 + y_n \\ y_{n+1} &= b x_n \\ \end{aligned}

Here, xnx_n and yny_n represent two variables, while aa and bb are parameters that control the chaotic behavior, with typical values being a=1.4a = 1.4 and b=0.3b = 0.3. The iteration number is represented by nn, and the Henon map begins with initial values x0x_0 and y0y_0 to generate the key. The process begins by generating a 128-bit key. This is followed by the creation of a bit stream that forms a random sequence. Then, the encryption process is initiated with a variable number of iterations, as illustrated in Figure 2.

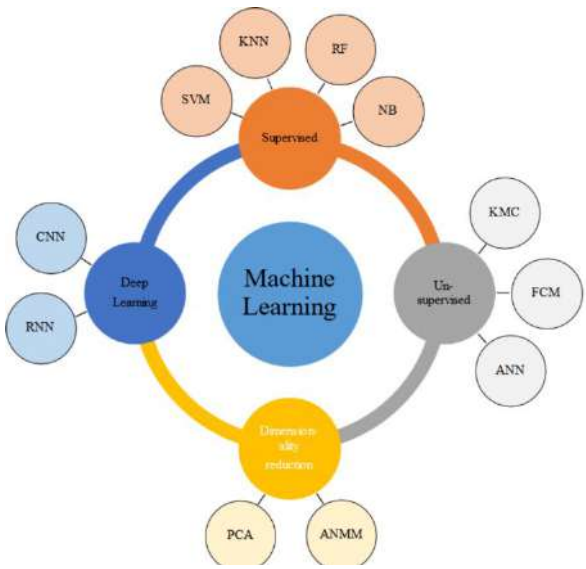

Figure 2: General illustration of proposed encryption system.

ISSN: 2393-9842
https://www.irjcs.com/archives

The described process outlines a general approach for image encryption using a deep neural network (DNN). Here's a more simplified version of the steps in the algorithm:

Figure 2: General Illustration of the Proposed Encryption System

This refers to a visual representation of how the encryption system functions, showing the main components and steps of the process.

Algorithm 1: General Proposed Stages for Image Encryption Using DNN

1. Read Images from Dataset

- Load the images that need to be encrypted from a collection or dataset.

2. For Each Image, Do the Following:

- 2.1 Preprocessing the Image:
  - Perform any necessary pre-processing on the image, such as resizing or normalization.
- 2.2 Extract Features from Image:
  - Extract key features or patterns from the image that will help in the encryption process.
- 2.3 Create Neural Network:
  - Define or initialize a neural network to be used in the encryption.
- 2.4 Determine Effective Parameters in the Network:
  - Identify the most important parameters or settings for the neural network to optimize its performance.

3. Update Hidden Layers and Nodes Based on Parameters:

- Adjust the neural network's internal parameters, such as weights and biases, to improve its capability in encryption tasks.

4. Confusion Process:

- 4.1 Use DNN to Select Image Partition:
  - Use the neural network to divide the image into smaller sections or partitions.
- 4.2 Scramble Each Partition Using DNN:
  - For each partition, apply scrambling techniques using the neural network to obscure the original image data.
- 4.3 Update Cipher Key:
  - Modify the encryption key as the image undergoes transformations.

5. Diffusion Process:

- 5.1 While Not End of Image (EOI):
  - Continue the process as long as the image hasn't been fully encrypted.
  - 5.1.1 Move Pixels into Vector:
    - Transform the image pixels into a vector format for easier processing.
  - 5.1.2 Use DNN to Alter Pixel Values (Vertically and Horizontally):
    - Use the neural network to change the pixel values in both vertical and horizontal directions.
  - 5.1.3 Update Cipher Key:
    - Again, modify the encryption key during this step to ensure further security.

6. Save or Transmit the Encrypted Image:

- Store the encrypted image in a file or send it securely for further use, such as processing or storage.

7. Return to Step 2 for Next Image:

- After encrypting one image, repeat the process for the next image in the dataset.

This process essentially combines techniques from deep learning (DNNs) and traditional cryptographic methods (confusion and diffusion) to create an encryption system that can adapt and secure image data effectively.

The deep learning algorithm utilizes the proposed method for encrypting medical images by first reading the images from a standard dataset. The entire process is then followed, which includes segmenting the images into blocks and modifying the pixel values, as described in Algorithm 1. To generate a high-quality encryption key, certain variables are introduced. These variables, designed to work in conjunction with deep learning, are explained as follows: Each layer of the neural network contributes to the encryption process. The encryption key is generated through a sequence of iterations, denoted as Ki=k1,k2,k3,…,k16K_i = k_1, k_2, k_3, \dots, k_{16}, which represents the initial cycles during which the hidden layers of the deep learning model are updated. The 16th key value (k16k_{16}) plays a critical role in creating a chaotic environment, further amplified by the structure of the nodes in the proposed neural network layers. The statistical behavior of this random system is depicted in Figure 3. The random number continually changes with each iteration, cycling through a random sequence, helping transform the number into the chaotic state of the system. A 128-bit key is employed, exhibiting highly random statistical behavior, as shown by the following equations:

In general, encryption consists of two stages. The first stage is confusion, where the pixel positions in the image are altered. These locations are of two types: First, the columns are adjusted according to a specific equation. The second step of the encryption process involves modifying the rows of the image to further enhance its randomness, increasing the security of the medical image. This alteration of rows and columns, combined with changes in the dependencies between parts of the image sliced from the original, boosts the complexity and security of the image through deep learning, as shown in Figure 4. The next stage of encryption, called diffusion, focuses on modifying the pixel values of the image, generating noise. To complete the encryption, an OR operation is performed between the pixel values, the encryption key (KK), and the scrambled image vector.

 https://doi.org/10.26562/irjcs

ISSN: 2393-9842
https://www.irjcs.com/archives

A key element of the encryption process is the random segmentation and division of the image. Additionally, the random distribution of pixels is achieved using deep learning, which involves multiple stages. These stages are based on the neural network's hidden layers, with a feedback mechanism that allows these layers to be reprogrammed based on the impact of each iteration. The core components of the deep neural network, including the input, hidden, and output layers, are shaped by deep learning, with a particular focus on the input layer in this work. Various parameters control the deep learning model, and some of these parameters are adjustable during the process. The desired outcome is achieved by using specific parameters that are fixed and can only be altered by modifying the structure of particular layers in the neural network. These parameters will be detailed in the discussion of the neural network design. Figure 5 illustrates the adaptive design of the deep neural network. In this context, ww represents the weight in the neural network, derived from each hidden layer. The hidden layers have variable lengths (i.e., the number of nodes) that pass information from one layer to the next. This relationship is expressed by the transition function:

ym=wnm·xmy_m = w_{nm} \cdotx_m

This function governs how information flows from one layer to the next (excluding recursive flow). The source node is given by xm=xm−1mod xmx_m = x_{m-1} \mod x_m, and the destination node is given by ym=ym+ym+1y_m = y_m + y_{m+1}. The key aspect here is controlling the recursive function, which will be discussed in the next section.

The output layer provides the final result of the neural network, reflecting the complexity of the desired outcomes. A hyperbolic equation models the result after multiple iterations in the neural stage to achieve accurate predictions. Additionally, a neural network can produce multiple different encryptions, so it is essential to automatically select the best one to optimize the system's efficiency. Therefore, machine learning, specifically deep learning, is proposed as a solution. The next section will elaborate on the contribution in this regard. Several variables control the neural network, and by training the network with these variables, the best possible prediction can be achieved. Various features are extracted from the medical image in the algorithm, with the most significant feature chosen and the least impactful ones disregarded. The process is repeated, and if the result improves, the system automatically adds another hidden layer. If the result stabilizes, one node is added to the specified hidden layer, and so on. The primary parameters of interest include weight, transition function, recursive number, flow, iteration ratio, feedback, and acknowledgment from each neuron and layer. To determine the most impactful parameter, the neural network structure must be fully integrated, ensuring that the network is completely connected to pass information through each stage. The standard setup of the neural system is represented by the infinite weighted summation of both current and past input signals (xj(n)x_j(n) and xj(n−1)x_j(n-1), respectively).

The output signal yj(n)y_j(n) is given by: yk(n)=∑i=0pwixj(n−i)y_k (n) = \sum_{i=0}^{p} w_ix_j(n-i)

where ww is the weight that controls the data flow within the network, and ii is the number of training iterations. The structure is shown in Figure 6. In this context, the weight derived from the structure generates an output signal yk(n)y_k(n) exponentially, which can be categorized into three cases: (a) w<1w < 1, representing a stable system; (b) w=1w = 1, indicating linear behavior; (c) w>1w > 1, suggesting a negative exponential. For practical purposes, the weight ww will be small enough to achieve a finite sum for yky_k. The flow of the system is another crucial parameter in the proposed method, as it directly impacts the results at the subsequent stage. Various cases can be considered for data flow, such as when a single input to a node produces one output, or when two inputs from different nodes are combined to produce a single output. For different iterations, the data flow may change and sometimes be classified into a Boolean value (accepted or not) at a given stage. The weight parameter (ww) acts as a bias or control factor for the system at each stage and node, influencing the system's behavior. This step estimates the neurons in the next stage based on the output function and the bias**,** which regulates the weight at each layer. Figure 7 demonstrates the behavior of the neural network during the update of multiple layers in the deep learning system, whether through forward or feedback data. The XOR operation is applied during iterations in various ways. The direction of the data flow may originate from an inherent node or from a recurrent flow, contributing to the system's overall performance. Figure 7 Strategy of node structure created and derivation. The factor for node to direct the new flow to the appropriate node. The last pattern must give achieve if (w_n+ w_n+$_1$) >0 then x_n+$_2$ = 1 and if ($x_n \oplus x_r(n+2)$) = 1 then new nod created. Two patterns at the same layer cannot achieve the same result function because of the consistent update and control of data flow by predicted bias. All transaction flow will store in certain vector to classify later as a deep learning system for better prediction. Recurrent network is the third parameter that can be used for deep neural network to increase accuracy. Recurrent network can be defined as the flow distinguishes from other feed forward neural layers and represented as at least one feedback loop.

RESULT AND DISCUSSION

In this section, the proposed encryption algorithm for securing medical images was thoroughly evaluated and tested. The study used two types of medical images: color and grayscale. Additionally, tests were performed on images from a standard dataset to benchmark the method's effectiveness and assess its strength. The image resolution used in the tests was 512 × 512, and the simulations were carried out using MATLAB 2015a on a laptop with a Core i9 CPU (3.9 GHz), 32 GB of RAM, and Fedora 32 as the operating system. The algorithm utilized a block size of 32, with partitioning set to n = 4**,** and the process was repeated 10,000 times. The proposed method is versatile and can be applied to images with various dimensions, as it operates on any pixel distribution and any sub-block partitioning. The segmentation process, it is not affected in any way, and the deep neural network works in the same way in different types of images.

SIMULATION RESULTS

Simulated results on image security refer to the use of computer simulations to assess the effectiveness of various techniques and algorithms designed to protect digital images. This process involves creating a digital image and then testing it against different types of attacks that could compromise its security, such as tampering, copying, or alteration. Common techniques used in simulated image security testing include watermarking; steganography, and encryption. In the proposed method, encryption is applied between two parties. Encryption is a technique that converts the image data into a form that can only be accessed or decrypted by authorized parties, thus preventing unauthorized access or alteration. Simulated results on image security are crucial for evaluating the performance of these techniques. They provide valuable insights into how well the security methods protect digital images from unauthorized access and tampering. In this study, various images were considered, most of which were sourced from a standard brain tomography dataset. Figure 8 depicts image labels that used in evaluation through proposed method within five classes. In encryption, there are many evaluation criteria achieved such as:

INFORMATION ENTROPY

The randomness of the image can be measured in information entropy and is defined by the following equation: For instance, $P(m)$ considers the probability of the appearance m, to grey scale image where the max entropy will be 8. When the entropy reaches 8, it means that the randomness of the image is large and good, that is, the distribution of pixels on the image is more random. Table 1 shows the randomness of the encrypted medical image.

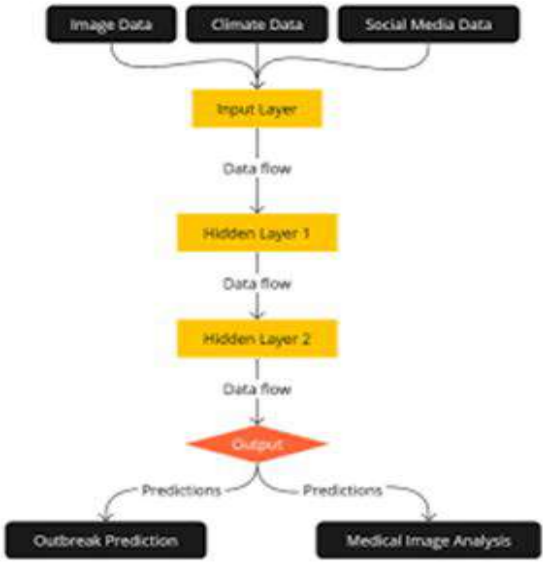

Figure 9 Histogram of pain image and encrypted image.

Table 2 Analysis of Chi-Square of encrypted image

Given Image	Encrypted Image
Image 1	261
Image 2	240
Image 3	231.8
Image 4	271
Image 5	212.5

IMAGE HISTOGRAM

The histogram shows how the pixels in an image are distributed. In the case of an encrypted image, the histogram should be altered to prevent attackers from inferring any information about the original image. The histogram of the encrypted image must differ significantly from that of the original image. Figure 9 illustrates the comparison of a medical image before and after encryption. The experiment also validated the histogram analysis of the encrypted image, with the calculations based on a square test, which is computed as follows: In this context, Oi represents the recurrence rate for the grey value i, and EV is the expected frequency of the grey scale, calculated as O/256. The Chi-Square value of the encrypted image is then displayed in Table 2Among the various tests used to evaluate the encryption algorithm; the Chi-Square test is particularly significant. It helps measure the similarity between the original and encrypted images. The expected encrypted image should resemble the original as closely as possible, and the Chi-Square test is a reliable method for this comparison. This statistical test is essential because it can differentiate between noise and encryption. Noise typically leads to the loss of image data, while encryption preserves all data intact. Thus, the Chi-Square test is a crucial metric for assessing the randomness of the encryption. In a standard image, adjacent pixels often exhibit a noticeable correlation. The strength of the encryption in an image can be measured by the extent to which this correlation is reduced between neighboring pixels. A lower correlation indicates a stronger encryption. The correlation between two adjacent pixels can be mathematically expressed.

ISSN: 2393-9842
https://www.irjcs.com/archives

In the equations above, A and B represent the values of two adjacent pixels, and s refers to the number of selected pixel pairs (A and B). The correlation coefficient values for the encrypted medical image are measured in vertical (V), horizontal (H), and diagonal (D) directions. In a well-encrypted image, the correlation coefficient should be close to zero. These values are shown in Table 3. The evaluation also includes a comparison of the proposed method with existing techniques in the literature, as presented in Table 4. Differential attacks involve making small changes to both the original and encrypted images, with the aim of guessing the information. The change is typically minimal, so as not to raise suspicion. To assess how well the encryption method performs against such attacks, two important metrics are used: the Number of Pixel Change Rate (NPCR) and the Unified Evaluation Method (UEM). These metrics help measure the effectiveness of the encryption by evaluating the sensitivity of the image to small alterations.

E1 and E2 represent two encrypted versions of an image: one from the original plain image and the other from a modified version (where a single pixel change is made to the plain image). The dimensions of the image are denoted by M (width) and N (height). To assess the strength of encryption against various attacks, two key metrics NPCR (Number of Pixel Change Rate) and UACI (Unified Average Changing Intensity) are used, with their values shown in Tables 5 and 6, respectively. The key space is a critical measure of an encryption system's robustness. A key space of at least 21002^{100} is necessary, as smaller key spaces would make the encryption vulnerable to brute-force attacks. In this algorithm, the initial condition Y0Y_0 and parameters like the control parameter aa and iteration number N0N_0 contribute to the key space. With Y0Y_0 having a value of 101610^{16} and N0=103N_0 = 10^3, the total key space becomes 103510^{35}, making the proposed algorithm highly resistant to brute-force attacks.

LIMITATIONS

Like all algorithms, the proposed one has limitations. The computational load is significant, as encryption, particularly for images, requires considerable time and resources. Key management is also a challenge; if the encryption key is lost or inaccessible, decryption becomes impossible. Future improvements should address recovery mechanisms for lost keys. Additionally, the algorithm faces compatibility issues with image compression or resizing, which may result in data loss, complicating the process.

CONCLUSION

Medical image security has become a crucial concern, prompting the use of AI algorithms to improve security beyond traditional methods. The proposed method leverages deep neural networks to segment medical images, randomly distribute blocks, and apply pixel-level randomness. This method enhances encryption through confusion and diffusion techniques, increasing randomness and scrambling pixel values. The deep neural network optimizes the weight calculations and feedback loops in hidden layers, which improves the encryption process. When evaluated against various criteria and compared with previous methods, the proposed algorithm demonstrates its effectiveness in securing medical images, proving to be a strong and efficient solution.

REFERENCES

1. https://pmc.ncbi.nlm.nih.gov/articles/PMC10388229/
2. https://www.mdpi.com/2504-4990/5/1/13
3. https://www.researchgate.net/publication/372825625_Secure_medical_image_transmission_using_deep_neural_network_in_e-health_applications
4. https://www.mdpi.com/2079-9292/11/1/157
5. https://www.google.com/search?sca_esv=41abccaec8fca31d&rlz=1C1RXQR_enGBIN1071IN1071&sxsrf=ADLYWIJclswGzoxoME2GMQTNbP37kaP8iA:1732209558097&q=Machine+learning+and+secure+image+transmission+for+disease+forecasting
6. https://www.google.com/search?sa=X&sca_esv=41abccaec8fca31d&rlz=1C1RXQR_enGBIN1071IN1071&udm=2&sxsrf=ADLYWII1LgB2IkYA8oOcCLoQbFtOmwzFOg:1732209687480&q=disease+prediction+using+machine+learning
7. https://pmc.ncbi.nlm.nih.gov/articles/PMC10662291/
8. https://www.nature.com/articles/s41598-023-31942-9
9. https://link.springer.com/article/10.1007/s11042-022-14305-w
10. https://ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/htl2.12049?af=R

