Expert Systems With Applications 252 (2024) 124287

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Image encryption algorithm based on DNA encoding and CNN

Kamlesh Kumar Raghuvanshi [a], Subodh Kumar [a], Sushil Kumar [b], Sunil Kumar [c,*]

[a] Department of Computer Science, Ramanujan College, University of Delhi, Delhi, India
[b] Department of Computer Science, Shyam Lal College(M), University of Delhi, Delhi, India
[c] Department of Instrumentation, Shaheed Rajguru College of Applied Sciences for Women, University of Delhi, Delhi, India

ARTICLE INFO

Keywords:
CNN
Permutation
Diffusion
DNA encoding
Bit-reversal

ABSTRACT

In this study, a novel, more stable, secure, and reliable image encryption model has been introduced. It combines a Convolutional Neural Network (ConvNet/CNN) model with an intertwining logistic map to generate secret keys. Additionally, initial conditions, control parameters, and secret keys are employed by the intertwining logistic map to produce diverse chaotic sequences. Permutation, DNA encoding, diffusion, and bit reversion operations are applied for scrambling and manipulating image pixels. The proposed encryption model was thoroughly examined using various analysis methods such as cropping attack, histogram analysis, key space evaluation, noise attack, information entropy assessment, differential attack, key sensitivity, and correlation coefficient examination. To expand the keyspace and enhance confusion and diffusion in the proposed encryption algorithm, the model employs different subkeys, private keys, and public keys through the Convolutional Neural Network. Furthermore, numerical and perceptual results were compared with the state-of-the-art outcomes to validate the model. Ultimately, the derived results demonstrate that the proposed intertwining logistic map-based image encryption model utilizing Convolutional Neural Network outperforms existing methods. This is due to its significant improvement in information entropy, enhanced randomness, high resistance against differential and statistical attacks, and overall efficiency.

1. Introduction

As a result of the implementation of various technologies and advances in information and communication technology, real-time data transfer and messaging have become increasingly important, especially for education, military, and medical industries. Generally, data is stored and transferred over open communication networks, which leads to various cyber threats such as phishing, network attacks, MITM (man-in-the-middle) attacks, and DoS (denial-of-service) attacks, among others. To transfer data securely over unsecured channels, the data should first be encrypted. It is clear that the design and development of an encryption model is a critical and essential task. Well-known text-based encryption models such as AES, DES, IDEA, and 3DES are widely used. However, these techniques have limitations and may not be suitable for multimedia applications such as images, due to the large amount of information and high correlation they contain (Wang et al., 2022b).

In the literature, various image encryption models based on spatial domains such as compressed sensing, DNA coding (Deoxyribonucleic Acid), cellular automata, and chaos, as well as frequency domains like FFT, DFT, and wavelet transforms, have been developed (Özkaynak, 2018; Sheng et al., 2022; Suri & Vijay, 2019; Wang & hui Wang, 2020; Yang et al., 2019; Yildirim, 2020). Chaos-based image encryption models have been widely used due to their properties such as sensitivity to initial conditions, high randomization, periodic doubling and bifurcation, complexity, expansivity, dense orbits, transitivity, and system parameters (Iqbal et al., 2021; Jasra & Hassan Moon, 2022; Vaziri et al., 2022; Wang & Su, 2021; Yildirim, 2022).

Deep learning models have gained widespread use in information security, particularly in applications such as image processing, image classification, and signal processing. For example, Li et al. (2018) applied convolutional neural networks (CNNs) to encrypt iris images optically. Chen et al. (2019) introduced a technique aimed at enhancing the resilience of 2D/3D optical image encryption through the use of an extended deep CNN. Ni et al. (2021) attempted to utilize a deep learning-based compressed sensing (CS) reconstruction algorithm for image encryption. Similarly, Liu et al. (2022) employed a deep learning-based bidirectional long short-term memory (BiLSTM) model alongside a block embedding technique for image encryption. Additionally, Wang et al. (2017) proposed an innovative encryption method for color images by leveraging DNA sequence operations and cellular neural network principles (Jun & Fun, 2021; Mohammed et al., 2022).

* Corresponding author.
E-mail addresses: raghukamlesh@gmail.com (K.K. Raghuvanshi), subodhkumar588@gmail.com (S. Kumar), kumar.sk106@gmail.com (S. Kumar), sunilkumar104@gmail.com (S. Kumar).

https://doi.org/10.1016/j.eswa.2024.124287
Received 27 September 2022; Received in revised form 2 May 2024; Accepted 19 May 2024
Available online 21 May 2024

Similarly, various image encryption models (Devipriya & Brindha, 2022; Jun & Fun, 2021; Ma et al., 2022; Wang et al., 2015, 2017; Zhang et al., 2023; Zheng et al., 2022) have incorporated deep learning and DNA operations, achieving satisfactory encryption outcomes. However, the full potential of the inherent characteristics of CNNs and DNA operations has yet to be completely explored, leaving room for further investigation by researchers (Basha et al., 2022; Bashir & Hanif, 2021; Chai et al., 2019; Xu et al., 2017).

Based on the preceding analysis, this paper introduces a color image encryption algorithm using DNA encoding and a convolutional neural network. This approach not only addresses the limitations of low security in low-dimensional chaotic systems and small key space but also leverages high storage density, minimal hardware and software resources, parallel processing, reduced computation, and low energy consumption.

1.1. Motivation of the paper:

The main motivation for using DNA rules, the intertwining logistic map, and a CNN-based approach is to make the model more resistant and enhance its security. DNA rules are applied to manipulate image pixel values (Jun et al., 2022). The intertwining logistic map is employed to encapsulate the nonlinear dynamic properties and serves as a diffusion model. Additionally, the use of CNN increases the model's sensitivity and resistance against various attacks. CNN also generates public keys, further improving security levels (Devipriya & Brindha, 2022).

In this document, we investigate an encryption model incorporating an intertwining logistic map, DNA encoding, and convolutional neural network. The model uses the intertwining logistic map to increase the randomness among image pixels, which improves information entropy. Additionally, it supports the model's resistance to various known attacks such as differential and statistical attacks. DNA encoding enhances the efficiency of the proposed encryption algorithm, while CNN expands the key space by incorporating public and private keys.

The image encryption algorithm takes less execution time because the permutation and diffusion processes are carried out in a single iteration. The performance of the encryption model is analyzed and evaluated using reliable metrics, including numerical values and visual representations. The obtained results indicate that the model is sensitive to both secret keys and data, as well as to bit reversion operations, and performs better than other CNN-based encryption models (Ma et al., 2022; Zhang et al., 2023; Zheng et al., 2022).

The model also conducts various analyses, such as brute-force attacks to determine keyspace, differential analysis by calculating NPCR and UACI scores, and statistical analysis involving information entropy analysis, histograms, and correlation coefficient analysis.

1.2. Contributions of the paper:

The primary contributions of the present work can be highlighted as stated below:

- A novel image encryption algorithm based on DNA encoding and CNN has been investigated.
- The proposed encryption algorithm encrypts test images at the pixel level. It requires fewer hardware and software resources due to the similarity in encryption and decryption structures.
- A two-tier shuffling process is introduced to enhance security levels and increase confusion. DNA rules and encoding modify the statistical properties of test images.
- An intertwining logistic map-based hyper-chaotic sequence is generated for use in a permutation mechanism that reduces the correlation among pixels.

Table 1
DNA encoding rules.

Rule	A(00)	C(01)	G(10)	T(11)
1	A	C	G	T
2	A	G	C	T
3	C	A	T	G
4	G	A	T	C
5	C	T	A	G
6	G	T	A	C
7	T	C	G	A
8	T	G	C	A

- A convolutional neural network-based method for generating a public key is proposed, making the model not only resistant to chosen or known plaintext attacks but also more sensitive to secret keys.
- The performance and security of the proposed encryption algorithm are assessed in detail, and the model proves effective in each run.

2. Theoretical principle

2.1. DNA rules and their encoding

Deoxyribonucleic Acid (DNA) is a biological term. It is a molecule which carries genetic codes to survive and to continue their generations. Adenine (A), Guanine (G), Cytosine (C) and Thymine (T) are nucleotides of DNA. According to DNA rules, each nucleotide is represented by two digits in binary numbers like A $\Rightarrow$ 00, C $\Rightarrow$ 01, G $\Rightarrow$ 10 and $T \Rightarrow 11$. Using DNA rules, one can represent a code with a total of twenty four combinations (Wang et al., 2015). In accordance with Watson-Crick complementary rule, twenty four combinations can be represented by Table 1, which displays eight encoding rules.

Eight 8-bits are used to represent each gray-scale pixel. It is equivalent to a DNA array having four nucleotides. For an example 10100111 is the binary array representation of pixel value 167. It is encoded into a number 00001110 as per rule-4 of encoding scheme. According to DNA nucleotides, it is also termed as AATG. It is to be noted from the rules discussed above that DNA encoding is a simple and effective technique. One can achieve secure encryption with the manipulation of image pixel values (Watson & Crick, 1953).

2.2. Convolution neural network

Artificial intelligence has played an important role in bridging the gap between machines and human capabilities. Research scientists are working in various domains, such as computer vision, to achieve remarkable advances. The primary goal in this field is to enable machines to perceive and understand the world as humans do, and then apply this knowledge to a range of tasks such as natural language processing, image and video recognition, recommendation systems, and image classification and analysis. The progress in computer vision, facilitated by the application of deep learning, has led to the development of convolutional neural networks (Geng et al., 2022).

Convolutional Neural Networks (CNNs) are specialized neural networks widely used for tasks such as image, audio, and speech recognition. CNNs consist of three primary types of layers: convolutional layers, pooling layers, and fully connected layers. Convolutional layers are central to the network, where inputs are convolved with adjustable-weight filters to extract features and form feature maps. Multiple convolutional layers can be stacked to refine features further. Pooling layers aggregate information from feature maps without adjustable weights, reducing complexity and mitigating overfitting.

Finally, the fully connected layer is crucial for classification. It uses filters applied to features and feature maps, along with an activation function, to generate predictions.

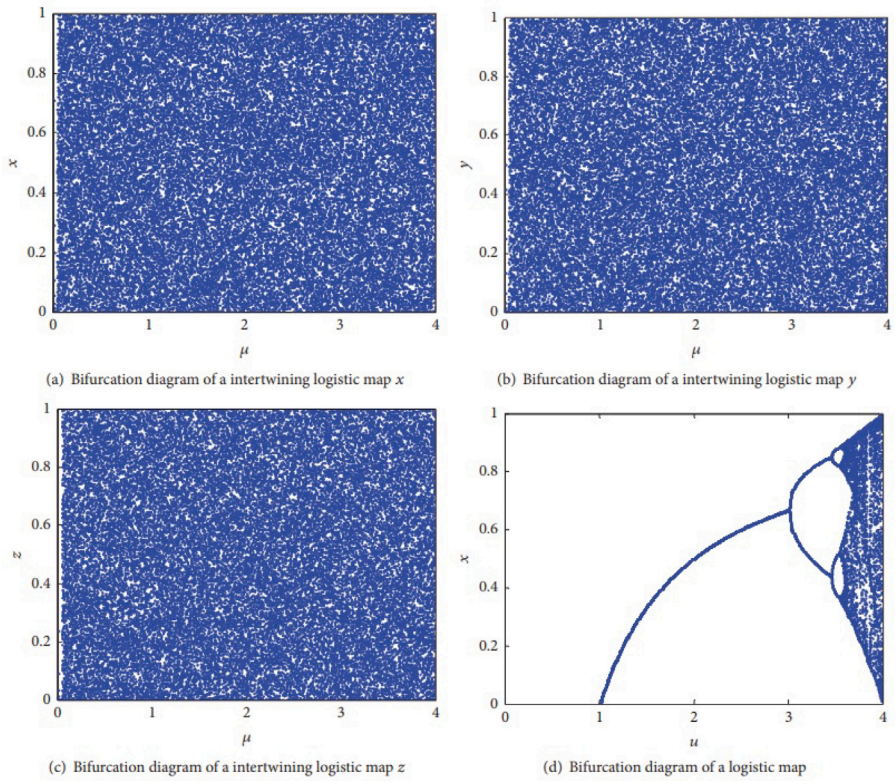


Fig. 1. Bifurcation diagram comparison of ILM and Logistic map.

CNNs excel in public key generation due to three main advantages: sparse interactions, parameter sharing, and equivariant representations. Traditional neural networks rely on matrix multiplication, resulting in a high parameter count and slower performance. In contrast, CNNs use smaller filters to detect meaningful features such as edges, reducing the number of parameters needed and enhancing efficiency. Moreover, CNNs share parameters across inputs, further minimizing hardware requirements.

In a Convolutional Neural Network (CNN), the operations conducted within convolutional layers can be broken down into distinct steps. The convolution operation is the central process within a CNN. It involves moving a filter across the input image and calculating the dot product between the filter and the overlapping region of the input at each position. Mathematically, this operation can be expressed as:

$$(f * g)(n) = \sum_{m} f(m) \bullet g(n-m)$$

Where: f is the input image, g is the filter, n is the position index, m is the index of the filter.

Within Convolutional Neural Networks (CNNs), each convolutional layer employs Rectified Linear Unit (ReLU) as its activation function. It is defined as:

$$f(x) = \begin{cases} x & x > 0 \\ 0 & x \le 0 \end{cases}$$

ReLU is more efficient than sigmoid or tanh functions, as it activates only some inputs and behaves linearly for positive values, simplifying optimization.

Pooling layers serve to diminish the dimensions of feature maps, thereby decreasing both the parameters requiring learning and the computational load within the network. After applying a pooling layer to a feature map with dimensions $n_h \times n_w \times n_c$, the resulting output dimensions are: $(n_h - f + 1)/s \times (n_w - f + 1)/s \times n_c$, n_h represents the height of the feature map, n_w denotes the width of the feature map, n_c signifies the number of channels in the feature map, f indicates the size of the filter, and s represents the stride length (Andono & Setiadi, 2022; Jain et al., 2021).

Finally, the fully connected layer executes two operations on the incoming data e.g. a linear transformation followed by a nonlinear transformation. Initially, a linear transformation is applied which is represented mathematically as follows:

$$Z = WT \bullet X + b$$

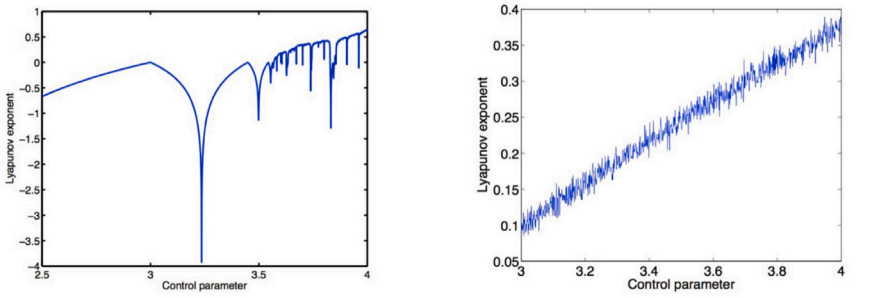
Where, X represents input, W represents weight, and b is a constant (bias). The linear transformation, by itself, is insufficient for capturing complex relationships. Hence, we introduce an additional element into the network to introduce non-linearity to the data. This additional component in the architecture is referred to as the activation function i.e. Sigmoid. Mathematically, it is represented in the following manner (Mohammed et al., 2022; Setiadi et al., 2022):

$$f(x) = \frac{1}{(1 + e^{-x})}$$

2.2.1. Performing feature extraction with VGG16

Feature extraction, a crucial process in pattern identification and visualization, involves representing data as numerical values. This step greatly contributes to improving model accuracy. Various techniques, including LDA (Linear Discriminant Analysis), PCA (Principal Component Analysis), and the local binary pattern technique, have been developed for effective feature extraction and transformation. Feature extraction divides and condenses large sets into smaller, more manageable groups, simplifying processing. The complexity of large datasets, with numerous variables, requires significant computing power for processing. To effectively decrease data size, the feature extraction process selects and combines variables to derive optimal features.

The VGG16 feature extraction model is renowned for its ability to extract a vast amount of data, resulting in high accuracy. It is particularly effective when applied in classification tasks using deep learning



(a) Lyapunov exponent of Logistic Map (b) Lyapunov exponent of Intertwining Logistic Map

Fig. 2. Lyapunov Exponent of Logistic map and Intertwining Logistic map.

models. Therefore, in the proposed work, the VGG16 model was chosen for feature extraction. Implemented on a dataset of ImageNet, the VGG16 model utilizes various layers within its architecture to derive features. These include fully connected layers, convolutional, pooling, and batch normalization layers, complemented by SoftMax & ReLU functions to construct the comprehensive model. VGG16 architecture, akin to the ImageNet model, comprises convolutional, fully connected, and pooling layers. There are 16 layers, eliminating the pooling layer. This architecture scales based on the layers count. The size of the input image for this particular network is consistently configured to 224 × 224 pixels, along with a filter size (3 × 3). In the concluding phase, the network incorporates an activation function that can offer class possibilities to the output(final) layers (Sharma et al., 2022).

2.3. Intertwining logistic map

Low dimensional chaotic maps are most commonly utilized in creating encryption models because of their uncomplicated structure and suitable properties. But these systems have limitations like small and weak key space, fewer control parameters, periodic behavior, starting point and stable & empty spaces(windows) observed within the divergence diagram.

In literature, it was observed that the attacker was able to find out the secret key easily (Ye & Huang, 2017). To overcome the aforesaid issues, higher-dimensional chaotic maps were used (Raghuvanshi et al., 2020). It leads to origin of intertwining logistic map (Kumar et al., 2018) and can be represented mathematically as follows:

$$X_{i+1} = [\mu \times K_1 \times Y_i \times (1 - X_i) + Z_i] \; MOD \; 1 \tag{1}$$

$$Y_{i+1} = [\mu \times K_2 \times Y_i + Z_i \times (1 + X_{i+1}^2)] \; MOD \; 1 \tag{2}$$

$$Z_{i+1} = [\mu \times (Y_{i+1} + X_{i+1} + K_3) \times Sin(Z_i)] \; MOD \; 1 \tag{3}$$

The value of μ should be greater than 0 but less than or equal to 3.999, whereas K_1, K_2 and K_3 should be greater than 34.9, 38.9 and 36.7 respectively. First 15000 iterations were discarded. It is to be noted from the equations $1-3$ that output of one equation is dependent on others. It makes equations more sensitive with respect to change even for one variable.

This is evident in the bifurcation diagram (Fig. 3) of the intertwining logistic map (ILM) and the logistic map. The diagram shows that the sequence generated by the ILM is uniform across the entire range, while the logistic map contains a blank window. The Lyapunov exponent represents the chaotic behavior of dynamical systems. If a system has at least one positive Lyapunov exponent, it is considered chaotic. Fig. 4 shows the Lyapunov exponent supporting the intertwining logistic map. It is evident that the ILM is random, complex, and suitable for use in cryptographic systems.

3. Proposed image encryption algorithm

3.1. Generating public keys from images through CNN based feature extraction

A CNN-based public key is generated using the VGG16 architecture. The key generation process using CNN is illustrated in Fig. 2. A pre-trained network is employed, consisting of datasets from ImageNet, which include one thousands object classes and a total of One million, two hundred eighty-one thousand, one hundred sixty-seven training images, Fifty thousand validation images, and One hundred thousand test images (Russakovsky et al., 2015). The Keras library's TensorFlow backend is used to implement this model (Capelo, 2018).

The CNN model utilizes pooling and convolution processes across five layers. To extract features from each image, a size of 7×7 with 512 channels is used, resulting in a total of $7 \times 7 \times 512 = 25{,}088$ features. These features are then transformed into a one-dimensional vector through the use of a flatten layer. Following the convolution and pooling layers, Dense-1 and Dense-2 layers are incorporated to further reduce the dimensions.

To adjust the initial weights of the networks, the Glorot-Uniform distribution is employed to randomly initialize the parameters of dense layers (Glorot & Bengio, 2010). Consider that the count of I/P (Input) and O/P(Output) within a layer is represented by $m_j \& m_{j+1}$, correspondingly.

It has been demonstrated that initializing the weights (Parameters) of a layer $\left(\mathrm{W} = \mathrm{U}\left[\pm\sqrt{\frac{6}{m_j + m_{j+1}}} \right]\right)$, uniformly leads to an output variance that closely approximates the input variance.

In the proposed model, a distinct public key is generated for that particular image in each run due to random initialization. The input variance is approximately equal to the output variance for the dense layer. In each dense layer, the activation function employed is the sigmoid function, denoted by (δ). It is assumed that F, D_1, and D_2 represent the outputs of the fatten, Dense-1, & Dense-2 layers, correspondingly. Bias initializers are configured to be 0. The output of dense layers is denoted as $D_1 = \delta(W_1 F)$ and $D_2 = \delta(W_2 F)$, where W_1 and W_2 represent the randomly initialized weights for D_1 and D_2, respectively. Before converting image features into a binary form, all the features are scaled between 0&1. The output from D_2 is compared with 0.5 for binarization purposes. In this case, a public key is acquired when ($P = D_2 > 0.5$) and the size is decided based on the dimension of the D_2 layer.

3.1.1. Comparison of key generation methods

Key generation is the most important part to develop image encryption model. In literature various encryption models are discussed. In the proposed work, we have generated public key based on CNN to be used in encryption. To confirm the superiority of the proposed model, we have computed various parameters like Uniformity Test, Robustness to

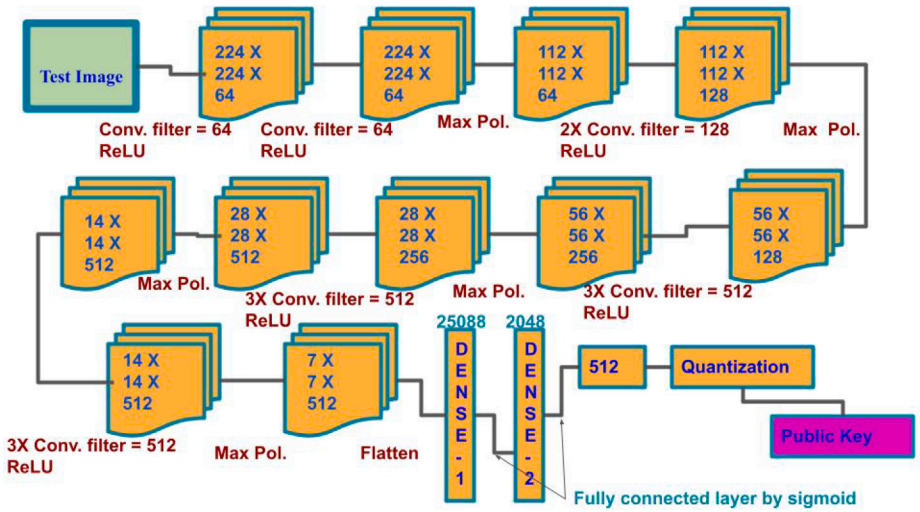


Fig. 3. Key generation using Convolution Neural Network.

Table 2
Comparison of key generation methods.

Type of test	DNA Based	BiLSTM	Proposed CNN
Entropy Analysis	3.875	3.545	4.0
Uniformity Test	Chi-square value: 0.5 P-value: 0.959	Chi-square value: 0.6 P-value: 0.966	Chi-square value: 0.3 P-value: 0.999
Robustness to Attacks	0.6625	0.4532	0.96875
Computation Efficiency	0.043 s	0.054 s	0.002 s
Generalization Test	Accuracy 0.4	Accuracy: 0.5	Accuracy: 0.95
Security Analysis	Entropy: 1.99	Entropy: 2.55	Entropy: 6.64
Cross-Validation score	0.47	0.26	0.95
Statistical Comparison	Accuracy: 0.45	Accuracy: 0.35	Accuracy: 0.96

Attacks, Computation Efficiency, Generalization Test, Security Analysis, Cross-Validation and Statistical as mentioned in Table 2.

The uniformity test validates the uniform distribution of keys. The average similarity between the original and perturbed CNN keys is 0.96875, indicating strong robustness against various known attacks. Additionally, it is observed that under the same hardware conditions, the time taken by CNN to generate secret keys is 0.002 s. Similarly, during the generalization test, the accuracy is measured at 95%, and the entropy of the CNN-generated keys is 6.64. Furthermore, cross-validation and statistical comparison scores show improvement. Based on the above analysis, it is evident that keys generated using CNN outperform other key generation methods like DNA and BiLSTM.

3.2. The proposed image encryption algorithm based on DNA encoding and CNN

Image encryption algorithm based on DNA encoding and CNN is depicted in Fig. 1. Bit reversion, public key using feature extraction, DNA encoding, Convolution neural network and intertwining logistic map is introduced. There are four stages i.e. permutation process, DNA encoding scheme, diffusion operation and bit reversion introduced to generate cipher. Public key and private keys are used to generate the main key used for encryption. To deduce chaotic sequences, control parameters & initial conditions are given as an input to a intertwining logistic map. These are followed with DNA encoding rules to generate ciphers.

3.3. Initial condition and various control parameters

In the proposed encryption model, CNN based public key is deduced as mentioned in Section 3.1. Various control parameters & initial conditions were obtained to deduce the chaotic sequences and further used to enlist public keys as well as secret keys. The generated chaotic sequences are also further used in different processes like bit reversion, permutation, diffusion and DNA encoding schemes. In this process of obtaining initial condition and control parameters, first an image matrix (A) is read followed by generation of fuzzy row matrix (B) based on CNN. Public key (C) is generated using column matrix and secret key (D) is generated using fuzzy row matrix. Main key (E) is deduced with the help of public key & secret keys and it is further divided into 8 subgroups. Each subgroup is the size of $8 \times 8 \times 8$. Each subgroup is further subjected to the modular operation and resulted into a binary matrix (F) of size 8×8. The generated binary matrix is further converted into a decimal matrix (S). At last, initial condition (V) and control parameters (U) are obtained using $(S_{1i} \ / \ 256)$ and $(S_{1i+4} \ / \ 256) \ + MOD \ 10$ respectively.

3.4. Permutation

Transposition technique is basically used to reposition the image pixels. Repositioning can be processed at different levels like bit, pixel or block. At bit levels, some bits in each pixel are permuted with the application of the permutation key. During both encryption & decryption processes at bit level, stacking and decomposition operations are utilized. With the help of simple logical operations, a perfect reconstruction is achieved.

In this operation, the first a chaotic sequence is produced based on initial conditions (V_1), control parameters (U_1) and a intertwining logistic map. The generated sequence is arranged in ascending order and the permutation matrix A_1 is obtained (Bigdeli et al., 2012).

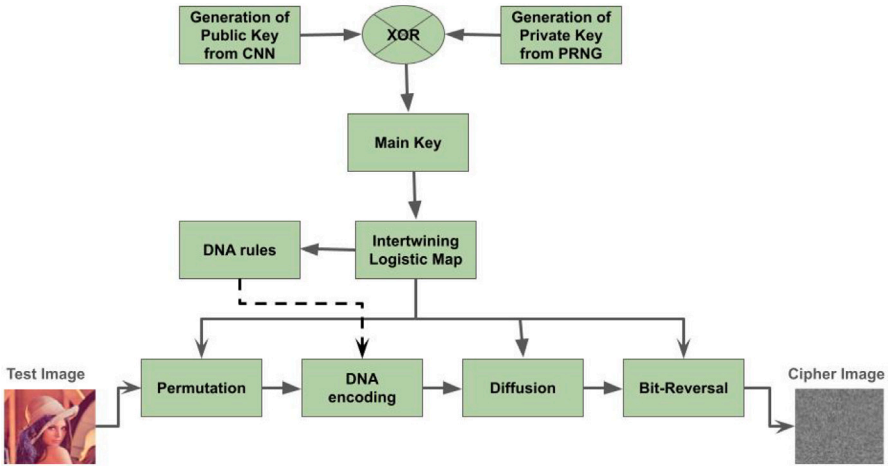


Fig. 4. Proposed encryption model.

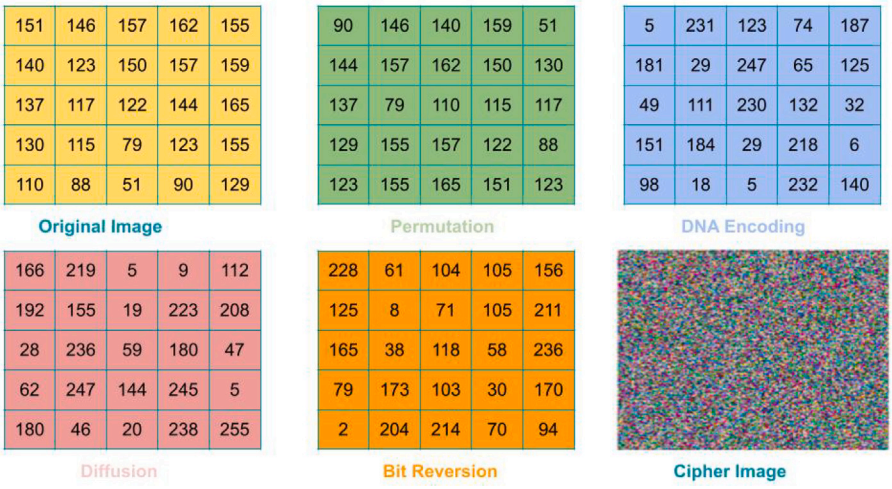


Fig. 5. Encryption process.

3.5. DNA encoding

In this operation, a second a chaotic sequence is produced based on initial conditions (V_2), control parameters (U_2) and intertwining logistic map. Then the fuzzy row matrix is obtained using the obtained chaotic sequence. The generated sequences are rearranged and converted into DNA encoding matrix A_2 based on DNA encoding rules.

3.6. Diffusion

Diffusion is the prominent method in each encryption technique. It holds pixels in a way that one pixel change affects more than 50% pixels. Original image has higher correlation and post diffusion it results in lower correlation among adjacent pixels. In this operation, third chaotic sequence is generated using initial conditions (V_3), x_1, x_2, x_3 and control parameters (U_3), μ, k_1, k_2 and k_3 and further used in intertwining logistic map based diffusion model. Scaling factor S_F was used to change the scale of the generated sequence. Then a fuzzy row matrix is deduced followed by a diffusion matrix A_3.

3.7. Bit reversion

In this operation, a third a chaotic sequence is produced based on initial conditions (V_4), control parameters (U_4) and intertwining logistic map. Then the fuzzy row matrix is deduced and arranged in reversed order followed by a Bit Reversion matrix A_4 (Erkan et al., 2022).

One can see from Fig. 6 (a-c) and Fig. 6 (d-f) cipher and recovered images (Lena & Baboon) respectively.

4. Results and analysis

CNN based secret key generation, DNA based permutation, intertwining logistic map based diffusion makes the model secure to transfer information over an unsecured network. Each method applied in the algorithm improvises the statistical and dynamical properties in a way to make it resistant to various well-known attacks such as chosen plaintext and known plaintext attacks, differential and statistical attacks. Validity of the model was undergone through various security findings and examination (Li, 2016). The proposed encryption algorithm has been executed using the Intel(R) platform, Core(TM) processor, $i5$ – 8265, Central Processing Unit 1.80 Gigahertz, Random Access Memory(RAM) = 4.00 GigaByte(GB), 64-Binary-digit(bit) OS(operating system), x64-based processor. Further, Windows 10 and MATLAB $R2022b$ have been used for the proposed model (see Fig. 5).

4.1. Keyspace analysis

All the possible permutation and combination of secret keys is termed as the key space. Brute force attack follows the same types

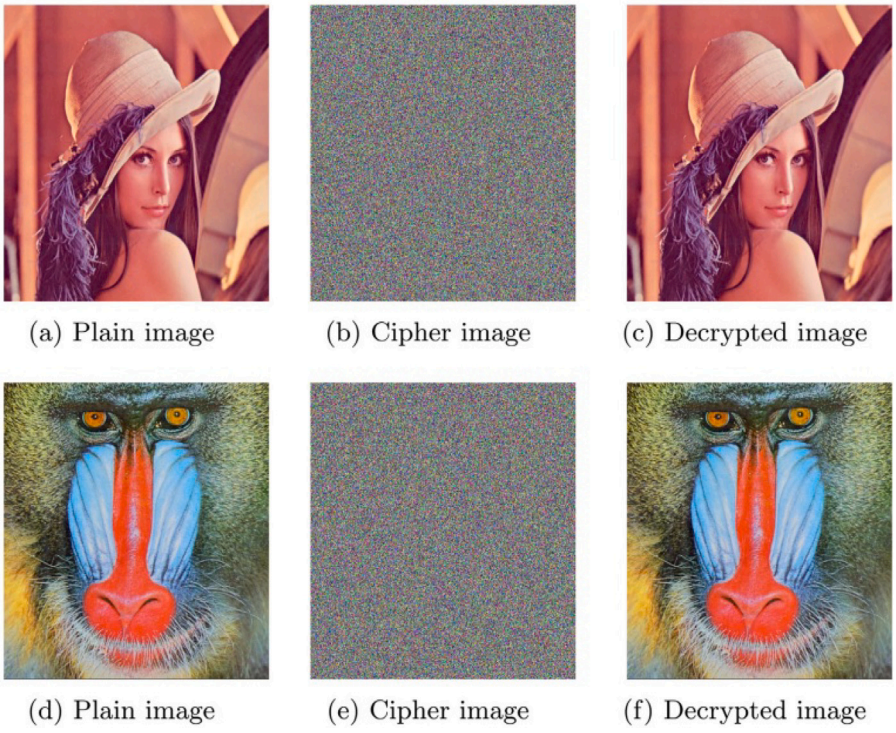


Fig. 6. Encrypting and decrypting the Lena & Baboon test images.

of key combinations to predict the secret keys. Shorter key sizes are vulnerable to brute-force attacks whereas a longer key size can resist it for a longer time. Hence, the key size should be optimal and difficult to guess. Brute force attacks are used to test the capability of the used secret key. A key is said to be secure and to be used in high security encryption against brute force attack if its size is longer than 2^{100} (Alvarez & Li, 2006). We are thankful to the reviewer's suggestion, We have calculated the Keyspace needed in the following way. In our proposed model, we have developed an approach using Convolutional Neural Networks to generate SHA-512. Using this key, we derive 16 floating-point numbers: $U_1, V_1, k_1, x_1, U_2, V_2, k_2, x_2, U_3, V_3, k_3, x_3, U_4, V_4, \mu$, and S_F with a precision of up to 10^{15}. These values serve as initial values and control parameters for encryption operations. The total keyspace is determined by multiplying the number of combinations in the first part by the number of combinations in the second part. This results in a keyspace of $10^{15\times16}$, which equals 10^{240} or approximately 2^{10240}.

4.2. Analysis of key sensitivity

An encryption model must exhibit high sensitivity concerning the secret key or data. A minor modification should outcome in a notable change in the test image. Several tests have been conducted to verify the key sensitivity. In this process, the original key with several false keys (one bit change in original key) were employed to encrypt the test image (Kumar et al., 2018).

Fig. 7 shows (a) Generated cipher with key K_1, (b) Generated cipher with key K_2, (c) Generated cipher with key K_3, (d) Generated cipher with key K_4 and (e) Generated cipher with key K_5. Where as Fig. 6 shows (f) Cipher difference using keys K_1 & K_2 (g) Cipher difference using keys K_1 & K_3 (h) Cipher difference using keys K_1 & K_4 (i) Cipher difference using keys K_1 & K_5. NPCR scores for the various images were computed, and the results for the test images are evaluated and presented in Table 3. NPCR scores of 99.6350%, 99.6245%, 99.6234%, and 99.6340% were computed for the key pairs (K_1 and K_2), (K_1 and K_3), (K_1 and K_4), and (K_1 and K_5), respectively. It can be concluded from the test results that altering just one bit in the secret key produces an entirely different cipher.

Table 3
NPCR score for the difference of ciphers.

Figure	Secret key	NPCR
Fig. 7 (f)	K_1&K_2	99.6350
Fig. 7 (g)	K_1&K_3	99.6245
Fig. 7 (h)	K_1&K_4	99.6234
Fig. 7 (i)	K_1&K_5	99.6340
Avg.	–	99.6292

Table 4
Hypothesis acceptance.

Metrics	Fig. 6 (b)	Fig. 6 (e)
p-value	0.3343	0.4275
Hypothesis (0 or 1)	0	0
Outcome	Accepted	Accepted

4.3. Analysis of histogram

A histogram is primarily used to depict the distribution of pixels in an image. Uniform pixel distribution makes the cipher image resistive against statistical or frequency attack (Arab et al., 2022).

The effect of CNN based key generation, permutation, DNA encoding, bit-reversion, various initial conditions and control parameter values used in intertwining logistic map can be seen in histogram analysis of Lena for red, green & blue channels (Fig. 8) and also same for Baboon images (Fig. 9). It is to be noted here that the original image has pixel distribution in a limited area whereas the generated cipher contains uniform pixel distribution. Therefore, the likelihood of extracting information from the cipher is minimal. Table 4 shows that values of $p > 0.05$, which indicates uniform cipher pixels distribution and also support the null hypothesis acceptance.

4.4. NIST randomness test

A data encryption model is said to be random in nature if it passes all the randomness tests given by NIST. NIST version $SP800-22$ is

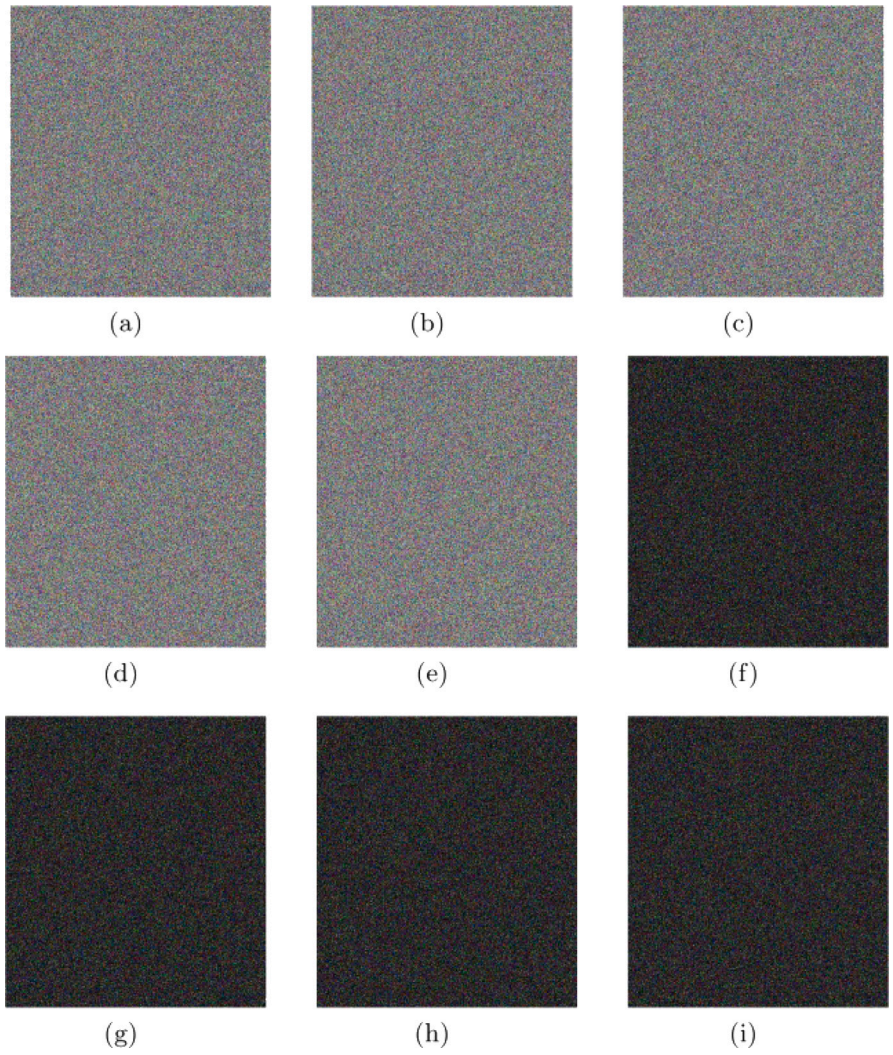


Fig. 7. Key sensitivity analysis.

the statistical package which runs 15 tests. All the computed $P-values$ are generated between 0 and 1. The generated sequences are said to be random if P-value $> \alpha$ (Predefined significant levels). Distribution of $P-values$ and Statistical tests are only two methods, suggested by NIST, to assess randomness in addition to uniform distribution. It is to be noted from Table 5 that $\alpha = 0.01$ (Significant Level) an individual component of the test image is considered a self-sufficient sequence. Further, generated sequence i.e. cipher found to be random and more secure (jun Gao et al., 2022).

4.5. Correlation analysis

Correlation coefficients make the relationship among adjacent pixels in an image. Its value falls between −1 & +1. Higher correlation is indicated by +1 in $+Ve$ direction whereas −1 indicates higher correlation in $-Ve$ direction among adjacent pixels. A good encryption model should generate a cipher whose adjacent pixels should have correlation coefficient value $\simeq 0$. One can see Correlation for original and cipher (Red, Green and Blue channels) image pixels in Fig. 10 for the different control parameter μ values of the Intertwining Logistic Map (Lai et al., 2022).

Mathematically, Correlation coefficient CC_{ab} is expressed as follows:

$$CC_{ab} = \frac{\mathrm{COV}(a,b)}{\sqrt{D(a) \times D(b)}} \tag{4}$$

$$E(a) = \frac{1}{M}\sum_{i=1}^{M} X_i \tag{5}$$

$$D(a) = \frac{1}{M}\sum_{i=1}^{M} (a_i - E(a))^2 \tag{6}$$

$$COV(a,b) = \frac{1}{M}\sum_{i=1}^{M} (a_i - E(a))\,(b_i - E(b)) \tag{7}$$

Here, M represents total image pixels & $i \in (1, \mathrm{M})$. As part of the correlation analysis in this study, $M = 3,000$ pixel samples were randomly selected from the cipher image. Table 6 shows the correlation coefficient value for Red channel adjacent pixels when ILM control parameter ($\mu = 3.7864$).

Symbols O_H-R, O_H-G and O_H-B (in Tables 6–8) constitute correlation coefficients in the test image for different channels in horizontal direction. Similarly, Symbols O_V-R, O_V-G and O_V-B (in Tables 6–8) constitute correlation coefficients in test images for different channels in vertical direction. Likewise, Symbols O_D-R, O_D-G and O_D-B (in Tables 6–8) constitute correlation coefficients in test images for different channels in diagonal direction.

Table 6 shows the correlation coefficient value for Green channel adjacent pixels when ILM control parameter ($\mu = 3.6989$). Likewise, Table 7 shows the correlation coefficient value for Blue channel adjacent pixels when ILM control parameter ($\mu = 3.8934$).

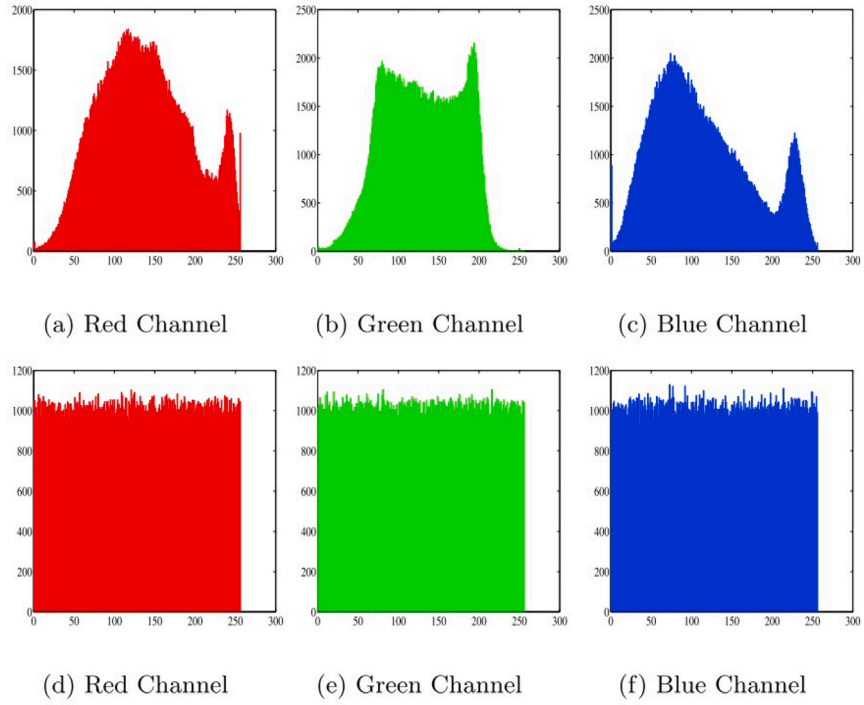


Fig. 8. Histogram analysis Lena image.

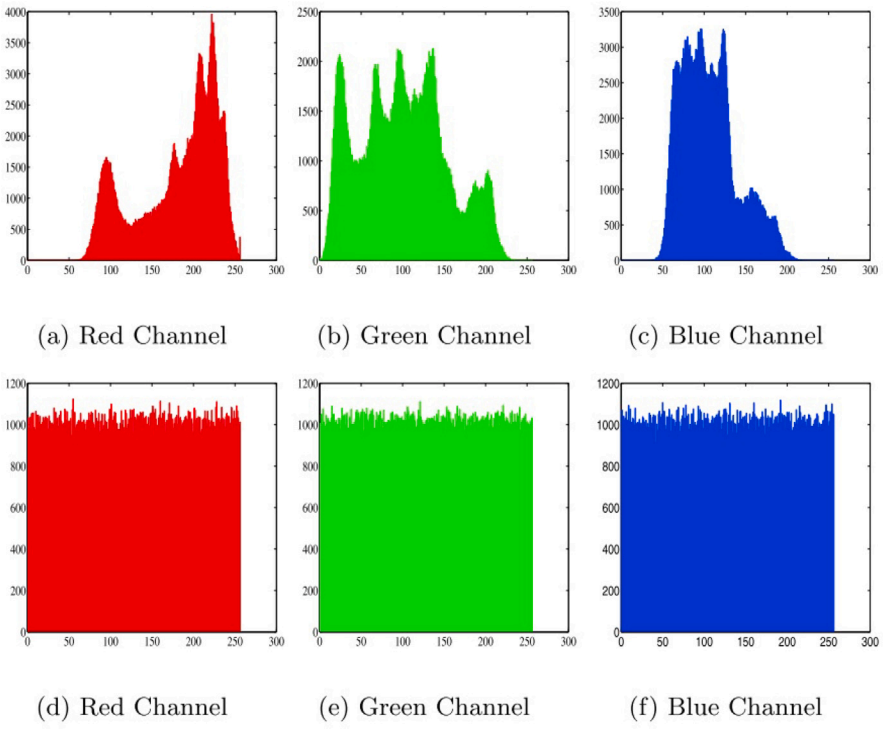


Fig. 9. Histogram analysis Baboon image.

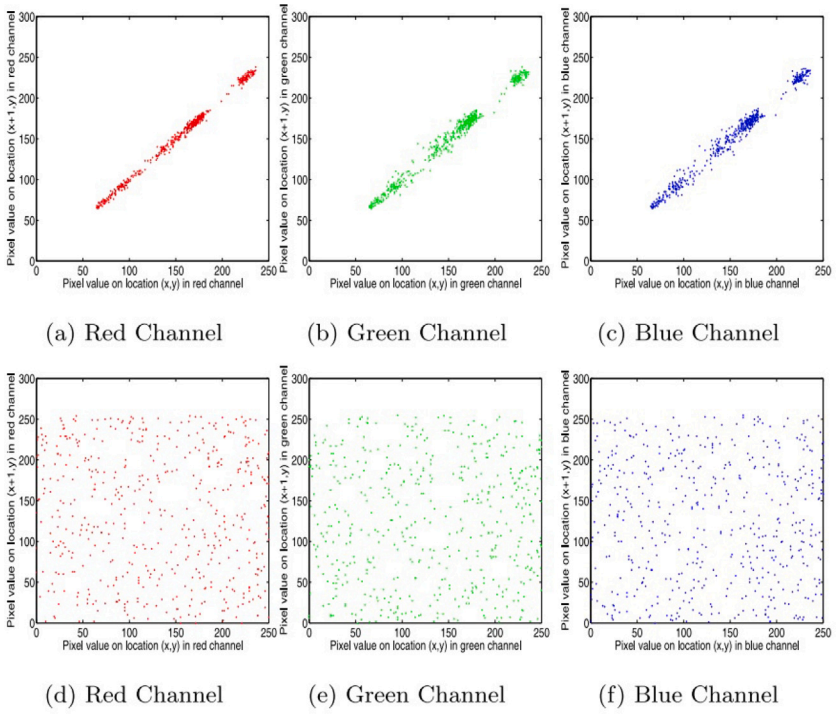


Fig. 10. Correlation for the Original image (a - c) and its respective cipher image (d - f). **Sample image: Lena image**.

Table 5
Randomness test.

Types of Test	P-Value	Outcome
Frequency Test	0.2813	Unpredictable in nature
Block Frequency Test	0.7845	Unpredictable in nature
Cumulative Sums (Forward) Test	0.3243	Random in nature
Cumulative Sums (Reverse) Test	0.1856	Unpredictable in nature
Runs Test	0.7587	Unpredictable in nature
Longest Run Test	0.0989	Unpredictable in nature
Rank Test	0.4498	Unpredictable in nature
FFT Test	0.0977	Unpredictable in nature
Non-overlapping Template (Average) Test	0.4845	Unpredictable in nature
Overlapping Template Test	0.2684	Unpredictable in nature
Universal Test	0.1689	Unpredictable in nature
Approximate Entropy Test	0.3997	Unpredictable in nature
Random Excursions (Average) Test	0.5432	Unpredictable in nature
Random Excursions Variant (Average) Test	0.3278	Unpredictable in nature
Serial (P-Value 1) Test	0.2698	Unpredictable in nature
Serial (P-Value 2) Test	0.2654	Unpredictable in nature
Linear Complexity Test	0.2879	Unpredictable in nature

Table 6
Correlation among pixels for varying μ.

Directions	Lena image	Boat image	Baboon image	Drop image	Pepper image	Tiffany image
$O_H - R$	0.9365	0.9344	0.9271	0.9709	0.9656	0.9897
$I_H - R$	0.0011	0.0015	0.0003	0.0014	−0.0054	−0.0031
$B_H - R$	−0.0012	0.0013	−0.0045	−0.0052	0.0007	0.0008
$C_H - R$	0.0034	−0.0032	0.0006	0.0018	0.0023	−0.0019
$O_V - R$	0.9256	0.9576	0.9689	0.9811	0.9968	0.9891
$I_V - R$	0.0012	0.0032	0.0024	−0.0022	0.0011	0.0002
$B_V - R$	0.0012	−0.0024	0.0022	0.0023	−0.0011	−0.0013
$C_V - R$	−0.0024	0.0003	−0.0011	−0.0012	0.0002	0.0004
$O_D - R$	0.9667	0.9456	0.9576	0.9721	0.9243	0.9656
$I_D - R$	−0.0020	0.0015	0.0005	−0.0023	−0.0021	−0.0022
$B_D - R$	0.0021	−0.0020	−0.0011	−0.0018	0.0003	0.0012
$C_D - R$	−0.0021	0.0011	0.0032	0.0033	0.0002	−0.0012

Table 7
Correlation among pixels for varying μ.

Directions	Lena image	Boat image	Baboon image	Drop image	Pepper image	Tiffany image
$O_H - G$	0.9431	0.9645	0.91265	0.9412	0.9543	0.9675
$I_H - G$	0.0011	0.0013	0 .0002	0.0003	−0.0013	−0.0012
$B_H - G$	0.0013	0.0011	−0.0014	0.0014	0.0004	0.0010
$C_H - G$	0.0011	−0.0013	0.0021	0.0019	0.0003	0.0022
$O_V - G$	0.9453	0.9613	0.9531	0.9912	0.9613	0.9943
$I_V - G$	0.0012	0.0011	0.0021	0.0011	0.0021	−0.0020
$B_V - G$	0.0011	−0.0012	0.0002	0.0014	−0.0016	−0.0013
$C_V - G$	−0.0011	0.0003	−0.0012	−0.0034	−0.0011	0.0012
$O_D - G$	0.9942	0.9432	0.9654	0.9678	0.9332	0.9245
$I_D - G$	−0.0012	0.0010	0.0015	−0.0011	−0.0030	−0.0021
$B_D - G$	0.0011	−0.0019	0.0011	−0.0022	0.0014	−0.0017
$C_D - G$	−0.0011	0.0017	0.0002	0.0003	−0.0021	−0.0013

Table 8
Correlation among pixels for varying μ.

Directions	Lena image	Boat image	Baboon image	Drop image	Pepper image	Tiffany image
$O_H - B$	0.9721	0.9465	0.9564	0.9787	0.9876	0.9675
$I_H - B$	0.0011	0.0012	0.0002	0.0012	−0.0013	−0.0011
$B_H - B$	−0.0013	0.0004	−0.0015	−0.0012	0.0013	−0.0021
$C_H - B$	0.0021	0.0012	0.0021	0.0017	−0.0019	0.0010
$O_V - B$	0.9567	0.9435	0.9546	0.9654	0.9672	0.9871
$I_V - B$	0.0011	0.0012	0.0023	0.0004	−0.0016	−0.0011
$B_V - B$	0.0012	−0.0010	0.0012	0.0019	−0.0016	−0.0019
$C_V - B$	−0.0013	0.0014	−0.0015	−0.0018	0.0011	0.0005
$O_D - B$	0.9987	0.9254	0.9456	0.9612	0.9754	0.9921
$I_D - B$	0.0003	0.0029	0.0005	0.0017	−0.0013	−0.0018
$B_D - B$	−0.0023	0.0021	−0.0001	0.0007	−0.0014	−0.0013
$C_D - B$	−0.0002	0.0013	0.0011	0.0012	0.0002	−0.0007

Table 9
Shannon entropy of different images for varying μ.

Directions	Lena image	Boat image	Baboon image	Drop image	Pepper image	Tiffany image
Original image	7.8781	7.7145	7.7652	6.4578	7.7587	7.7579
ILM (μ = 3.7864)	7.9991	7.9998	7.9989	7.9992	7.9991	7.9995
ILM (μ = 3.6989)	7.9996	7.9994	7.9997	7.9995	7.9994	7.9996
ILM (μ = 3.8934)	7.9993	7.9995	7.9994	7.9997	7.9995	7.9998

4.6. Analysis of information entropy

Information Entropy Analysis is employed to compute the randomness of generated cipher image. Information entropy (H_{IE}) was defined by Shannon for a given symbol sl_i as follows (Teng et al., 2022):

$$H_{IE} = -\sum_{i=0}^{2^{N-1}} I(sl_i)\log_2 I(sl_i) \tag{8}$$

Here, symbol $I(sl_i)$ denotes the probability to find symbol sl_i. The computed values of Shannon Entropy (Table 9) are found to be constant approximately for cipher images when control parameter μ is varied. It is to be noted from Table 8 that even for varying μ, information entropy was found to be approximately 8 which supports the model's randomness and no information leakage.

4.7. Analysis for differential attacks

Unified average change intensity (UACI) & Number of pixels change rate (NPCR) scores are utilized in the analysis of differential attack. To compute both the NPCR and UACI scores, two sample images (Im_1 and Im_2) that differ by one pixel are encrypted to generate ciphers. The difference between both the ciphers are computed to find-out NPCR and UACI scores. W (Width) × H (Height) denotes the size of Gray scale test images $Im_1(k,l)$ and $Im_2(k,l)$ (Huang & Zhou, 2022). If $Im_1(k,l) = Im_2(k,l)$ then $Zm(k,l) = 1$ otherwise $Zm(k,l) = 0$.

Here $Zm(k,l)$ is $3rd$ array and $k \in W$ & $l \in H$.

$$NPCR = \frac{\sum_{k,l} Zm(k,l)}{W \times H} \times 100\% \tag{9}$$

$$UACI = \frac{\sum_{k,l} \frac{Im_1(k,l) - Im_2(k,l)}{255}}{W \times H} \times 100\% \tag{10}$$

Table 10
NPCR score for different μ.

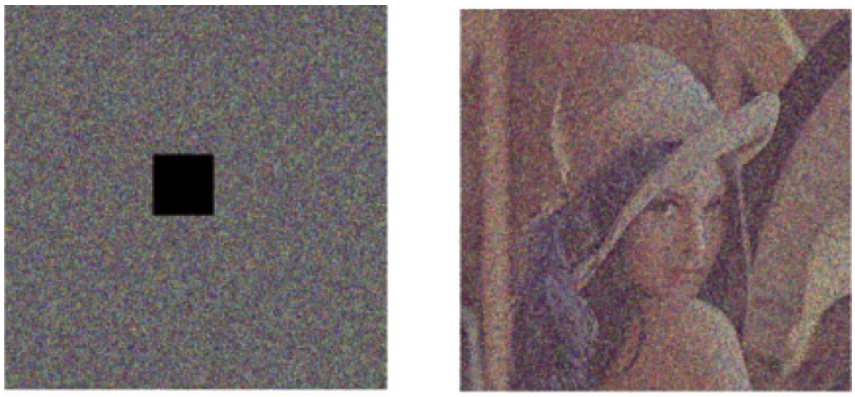
Image	ILM (μ = 3.7864)	ILM (μ = 3.6989)	ILM (μ = 3.8934)	Xu et al. (2017)	Basha et al. (2022)
Lena image	99.73	99.76	99.72	99.73	99.61
Boat image	99.70	99.71	99.74	99.72	NA
Baboon image	99.75	99.78	99.74	99.72	NA
Drop image	99.73	99.71	99.72	NA	NA
Pepper image	99.71	99.69	99.67	99.70	99.63
Tiffany image	99.66	99.68	99.69	NA	NA

Table 10 shows NPCR score for varying control parameters μ = 3.7864, (μ = 3.6989), (μ = 3.8934) and in comparison to methods discussed in literature for image encryption.

Table 11 shows socre of UACI for varying control parameters μ = 3.7864, (μ = 3.6989), (μ = 3.8934) and also compared with existing methods for image encryption.

4.8. Analysis of cropping attack

Analysis of Cropping attack is carried out to check the robustness of any model. Here, a group of pixel values are changed with 0s. Cropping



(a) Cropped cipher image (b) Retrieved image after cropping

Fig. 11. Cropped test.

Table 11
UACI score for different μ.

Image	ILM ($\mu = 3.7864$)	ILM ($\mu = 3.6989$)	ILM ($\mu = 3.8934$)	Xu et al. (2017)	Basha et al. (2022)
Lena image	33.36	33.30	33.32	37.57	33.53
Boat image	33.33	33.31	33.38	37.67	NA
Baboon image	33.39	33.41	33.32	28.19	NA
Drop image	33.33	33.30	33.39	NA	NA
Pepper image	33.31	33.40	33.42	26.58	33.58
Tiffany image	33.36	33.38	33.49	NA	NA

attack take places on data to be transmitted over the communication network. Encrypted image is snipped for a 100 × 100 window. From Fig. 11(a), one can see that the cropped image in Fig. 11 can be easily identified after recovery in Fig. 11(b). A PSNR value of 28.9276 db, calculated after analysis, demonstrates the encryption algorithm's resistance to known and cropping attacks (Basha et al., 2022).

4.9. Compression

Multimedia information i.e. images contain bulk information, so compression is required to transfer data over the communication network. To compress data, encoding is the most commonly used method. In the proposed encryption algorithm, Lossless compression method (JPEG) is used. As shown in the Fig. 12, encrypted image (a) is compressed for various compression ratios like Fig. (b) CR ≥ 4.9971, Fig. (c) CR ≥ 8.8675, Fig. (d) CR ≥ 12.8965, Fig. (e) CR ≥ 15.8453 and Fig. (f) CR ≥ 28.7965. Once the compression ratio is increased ≥ 28.7965, the generated image is distorted completely. The compression test results indicate that the proposed encryption algorithm is robust against lossless compression (Jain et al., 2021).

4.10. Noise attack analysis

Cyber attacks can insert or add some noise naturally during data transfer over a communication network. Noise attack analysis is basically used to deduce permutation as well as manipulation metricizes for an encryption model like crop attack analysis. To evaluate the proposed encryption model, Salt & pepper noise with varying densities (0.002, 0.006, 0.02, & 0.1) were incorporated into the cipher image. Further, cipher image with added noise was recovered with the decryption technique and corresponding Peak Signal to Noise Ratios i.e. 41.13 db, 31.72 db, 28.14 db, and 18.45 db respectively were obtained. The test results (Fig. 13) indicate that the proposed image encryption algorithm successfully restores all test images for salt and pepper noise attack densities up to 0.1 db (Wang et al., 2022).

4.11. Computational complexity

Security is the most important feature of a Cryptographic system. On the other hand, if an encryption algorithm takes comparably shorter running time, said to be better for specific applications in particular environments. The proposed encryption algorithm has been executed using the Intel(R) platform, Core(TM) processor, $i5-8265$, Central Processing Unit 1.80 Gigahertz, Random Access Memory(RAM) = 4.00 GigaByte(GB), 64-Binary-digit(bit) OS(operating system), x64-based processor. Further, Windows 10 and MATLAB $R2022b$ have been used for the proposed model. Theoretical and empirical approaches are the most common methods used to calculate computational complexity. In case of empirical method, Computation time (Running time) is calculated from the time required to execute the algorithm. Even though it looks better and outstanding, it does not render the inherent outcome of an algorithm. The reaction time rely upon various factors like specific input, certain compiler, specific hardware and software etc. as it affects the performance of the algorithm. Table 11 presents a comparison of the encryption speed of the proposed encryption algorithm with other cryptographic models. It is important to note that the proposed encryption model performs better.

Encryption Throughput (ET) is another metric to calculate the performance of a cipher. It provides the density or amount of image data to be encoded per unit of time. Mathematically, It is calculated as follows (Eqn. 11):

$$Encryption\ Throughput\ (ET) = \frac{Image_{size}(Bits)}{Encryption_{Time}(Seconds)} \tag{11}$$

In Table 12, the last column shows the value of Encryption throughput. The proposed model performs better for speed and ET(average value). In general, Asymptotics (Theory of mathematics) is used to analyze the algorithms (Li, 2004). In the case of image cipher, the procedure to calculate chaotic parameters, steps for encryption algorithms used, play an important role to deduce the time complexity. After iterating intertwining logistic map, $\Theta(27mn)$ cost was calculated to get nine streams, image matrix (A), fuzzy row matrix (B), Public key (C), secret key (D), Main key (E), binary matrix (F), decimal matrix (S), initial condition (V) and control parameters (U) for generated random number each having size of $3mn$. After converting these parameters to get

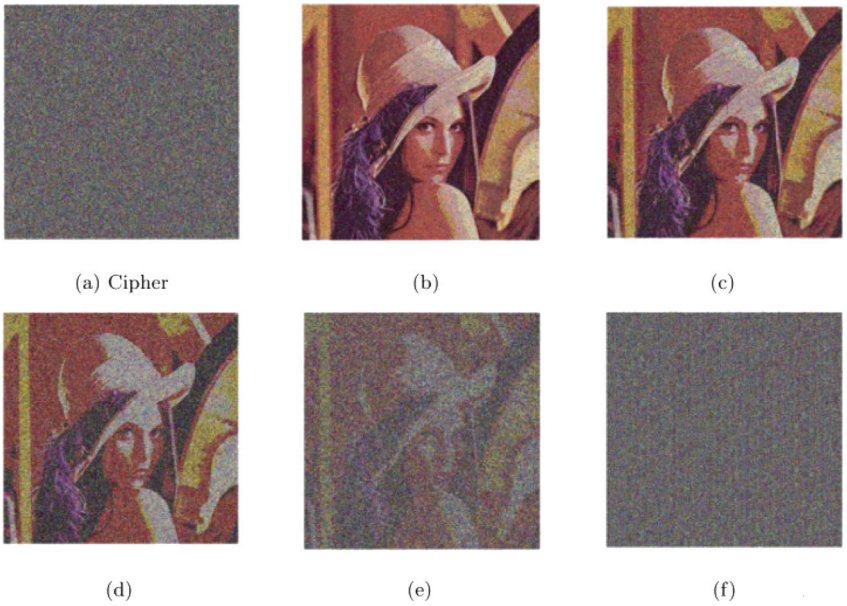


Fig. 12. CR Test Analysis for Lossless medium. (a) Generated Cipher image, (b) CR ≥ 4.9971, Fig. (c) CR ≥ 8.8675, Fig. (d) CR ≥ 12.8965, Fig. (e) CR ≥ 15.8453 and Fig. (f) CR ≥ 28.7965.

Fig. 13. Cipher images with varying levels of added salt and pepper noise densities (a) 0.002, (b) 0.006, (c) 0.02, (d) 0.1 & corresponding recovered images for same Salt & Pepper Noise densities (e) 0.002, (f) 0.006, (g) 0.02, (h) 0.1.

Table 12
A comparison of the encryption speed between the proposed encryption algorithm and other cryptographic models.

Types of Algorithm	Test image	Speed (s)	Mbit/s
Proposed Model	Lena image	3.0690	0.1414
	Baboon image	3.0112	0.2134
	Pepper image	3.0081	0.2743
	Boat image	3.1031	0.2156
	White image	3.0806	0.0018
	Black image	3.1023	0.0012
	Average	**3.0623**	**0.1412**
Chai et al. (2019)	Lena image	NA	1.28
Bashir and Hanif (2021)	Lena image	3.1143	0.1707

Table 13
A Comparison of Proposed encryption algorithm with other encryption models for different parameters.

Algorithm	Correlation coefficient			Entropy	NPCR	UACI
	R	G	B			
Proposed model	0.0011	0.0011	0.0002	7.9996	99.76	33.36
Wang et al. (2017)	−0.0219	0.0326	−0.0098	7.9914	99.45	33.28
Liu et al. (2022)	−0.0046	0.0072	0.0009	7.9916	–	–

Public key, private key and DNA Rules, repeatedly the cost calculated is $\Theta(9mn)$. The cost calculated for the pixel level Permutation and DNA Encoding is $\Theta(9mn)$. The cost estimated for diffusion at the DNA level and bit reversal is $\Theta(9mn)$. Hence, the total cost calculated for the complete process is $\Theta(54mn)$.

4.12. Comparisons

In this subsection, we have considered different parameters such as NPCR and UACI scores, entropy and correlation coefficient in comparison with other encryption models.

Table 13 depicts the comparison of Proposed model with other models for different parameters. Therefore, the proposed encryption model outperforms other image encryption algorithms.

4.13. Analysis of chi square

It is a statistical method utilized to assess the consistency or conformity of the histogram of the image. It represents the distribution of pixel frequencies within an image, displayed graphically. The image values span from 0 to 255, as evident in the x-axis of the histogram showcased in Fig. 9. In a typical image histogram, tonal variations emerge from the distribution of pixels in the image. In contrast, histogram of an encrypted image should exhibit consistent tonal variations across pixels, with values ranging from 0 to 255. As the histogram solely offers a pictorial representation, conducting a χ^2 analysis becomes crucial to furnish substantial verification. The χ^2 analysis can be computed using Equation 12. Here, V_i represents the iteration value of gray level i, & F signifies the frequency calculated for the value of each gray ($F = V/255$).

$$\chi^2 = \sum_{i=1}^{256} \frac{(V_i - F)^2}{F} \tag{12}$$

As a point of clarification, the value of i varies between 1 to 256, as indexing in MATLAB begins at 1.

Table 14
Measurement and comparison of the outcomes of χ^2 test analysis.

Image	Color/ Black & white	Mohammed et al. (2022)	Jun and Fun (2021)	Proposed
Lena image	Color	–	–	252.1823
Baboon image	Color	264.1074	–	245.3215
Airplane image	Color	259.4973	–	248.1324
Peppers image	Color	260.8383	–	240.1124
Splash image	Color	271.3498	–	252.2165
House image	Color	258.5586	–	272.3456
Tree image	Color	–	–	248.63
Jelly Beans image	Color	–	–	260.1578
Lena image	Black & white	–	274.7188	245.3023
Baboon image	Black & white	–	275.7813	258.3212
Peppers image	Black & white	–	261.1406	238.4132
Boat image	Black & white	–	278.1641	242.8914
Average		262.8703	272.4512	250.3355
Pass Rate		5/5	4/4	12/12

Using a significance level δ of 0.04 and a degree of freedom (df) of 255, the calculated chi-square value ($\chi^2_{(\delta,df)}$ equals 289.3589. Hence, when the obtained χ^2 value is lower, it signifies confirmation that the histogram demonstrates uniformity (Setiadi et al., 2022).

The χ^2 analysis findings are outlined in Table 14, including a comparison with prior research that used the same dataset. Additionally, grayscale images were tested to broaden the scope of comparison. According to the proposed chi-square measurement tool, the method demonstrates superiority and successfully meets the criteria for uniformity (Andono & Setiadi, 2022).

5. Conclusion

In this work, a cryptographic model is proposed, depending on a new diffusion via Intertwining logistic map based, permutation via DNA encoding scheme, key generation via random number generator, Convolution Neural Network & a bit reversal operation. To manipulate, scramble, sensitive and to reduce correlations among image pixels, permutation process, DNA encoding scheme, diffusion model and bit reversion techniques have been introduced respectively. A pseudo-random number generator (PRNG), public and private keys are used to generate various parameters and further used as an input by intertwining logistic map to generate diverse hyper-chaotic sequences. Performance evaluation of the proposed encryption model was implemented both numerically i.e. key space analysis, NIST, NPCR & UACI, Information entropy and visually i.e. Noise attack analysis, cropped test, correlation analysis, histogram analysis. Further, results were validated by comparing other image encryption models in literature. Therefore, the obtained results suggest that the proposed encryption algorithm is most appropriate among other image encryption models and also best suited for real time multimedia applications.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

CRediT authorship contribution statement

Kamlesh Kumar Raghuvanshi: Conceptualization, Investigation. **Subodh Kumar:** Writing and editing. **Sushil Kumar:** Validation, Coding. **Sunil Kumar:** Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

Authors acknowledges the support provided by University of Delhi, India.

References

Alvarez, G., & Li, S. (2006). Some basic cryptographic requirements for chaos-based cryptosystems. *International Journal of Bifurcation and Chaos*, *16*(08), 2129–2151. http://dx.doi.org/10.1142/S0218127406015970.

Andono, P. N., & Setiadi, D. R. I. M. (2022). Improved pixel and bit confusion-diffusion based on mixed chaos and hash operation for image encryption. *IEEE Access*, *10*, 115143–115156. http://dx.doi.org/10.1109/ACCESS.2022.3218886.

Arab, A. A., Rostami, M. J. B., & Ghavami, B. (2022). An image encryption algorithm using the combination of chaotic maps. *Optik*, *261*, Article 169122. http://dx.doi.org/10.1016/j.ijleo.2022.169122, URL: https://www.sciencedirect.com/science/article/pii/S0030402622004843.

Basha, S. M., Mathivanan, P., & Ganesh, A. B. (2022). Bit level color image encryption using logistic-Sine-tent-Chebyshev (LSTC) map. *Optik*, *259*, Article 168956. http://dx.doi.org/10.1016/j.ijleo.2022.168956, URL: https://www.sciencedirect.com/science/article/pii/S0030402622003369.

Bashir, Z., & Hanif, I. N. (2021). A novel gray scale image encryption scheme based on pixels' swapping operations. *Multimedia Tools and Applications*, *80*, 1029–1054. http://dx.doi.org/10.1007/s11042-020-09695-8.

Bigdeli, N., Farid, Y., & Afshar, K. (2012). A robust hybrid method for image encryption based on hopfield neural network. *Computers & Electrical Engineering*, *38*(2), 356–369. http://dx.doi.org/10.1016/j.compeleceng.2011.11.019, URL: https://www.sciencedirect.com/science/article/pii/S0045790611001959.

Capelo, L. (2018). *Beginning application development with tensorflow and Keras: Learn to design, develop, train, and deploy TensorFlow and keras models as real-world applications*. Packt Publishing.

Chai, X., Fu, X., Gan, Z., Lu, Y., & Chen, Y. (2019). A color image cryptosystem based on dynamic DNA encryption and chaos. *Signal Processing*, *155*, 44–62. http://dx.doi.org/10.1016/j.sigpro.2018.09.029, URL: https://www.sciencedirect.com/science/article/pii/S016516841830313X.

Chen, J., Li, X.-W., & Wang, Q.-H. (2019). Deep learning for improving the robustness of image encryption. *IEEE Access*, *7*, 181083–181091. http://dx.doi.org/10.1109/ACCESS.2019.2959031.

Devipriya, M., & Brindha, M. (2022). Image encryption using modified perfect shuffle-based bit level permutation and learning with errors based diffusion for IoT devices. *Computers & Electrical Engineering*, *100*, Article 107954. http://dx.doi.org/10.1016/j.compeleceng.2022.107954, URL: https://www.sciencedirect.com/science/article/pii/S0045790622002300.

Erkan, U., Toktas, A., Enginoğlu, S., Akbacak, E., & Thanh, D. N. H. (2022). An image encryption scheme based on chaotic logarithmic map and key generation using deep CNN. *Multimedia Tools and Applications*, *81*(5), 7365–7391. http://dx.doi.org/10.1007/s11042-021-11803-1.

jun Gao, Y., wei Xie, H., Zhang, J., & Zhang, H. (2022). A novel quantum image encryption technique based on improved controlled alternated quantum walks and hyperchaotic system. *Physica A. Statistical Mechanics and its Applications*, *598*, Article 127334. http://dx.doi.org/10.1016/j.physa.2022.127334, URL: https://www.sciencedirect.com/science/article/pii/S0378437122002655.

Geng, Z., Li, J., Han, Y., & Zhang, Y. (2022). Novel target attention convolutional neural network for relation classification. *Information Sciences*, *597*, 24–37. http://dx.doi.org/10.1016/j.ins.2022.03.024, URL: https://www.sciencedirect.com/science/article/pii/S0020025522002237.

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In Y. W. Teh, & M. Titterington (Eds.), *Proceedings of machine learning research: vol. 9, Proceedings of the thirteenth international conference on artificial intelligence and statistics* (pp. 249–256). Chia Laguna Resort, Sardinia, Italy: PMLR.

Huang, Z.-W., & Zhou, N.-R. (2022). Image encryption scheme based on discrete cosine Stockwell transform and DNA-level modulus diffusion. *Optics and Laser Technology*, *149*, Article 107879. http://dx.doi.org/10.1016/j.optlastec.2022.107879, URL: https://www.sciencedirect.com/science/article/pii/S0030399222000366.

Iqbal, N., Hanif, M., Abbas, S., Khan, M. A., & Ul Rehman, Z. (2021). Dynamic 3D scrambled image based RGB image encryption scheme using hyperchaotic system and DNA encoding. *Journal of Information Security and Applications*, *58*, Article 102809. http://dx.doi.org/10.1016/j.jisa.2021.102809, URL: https://www.sciencedirect.com/science/article/pii/S2214212621000491.

Jain, K., Aji, A., & Krishnan, P. (2021). Medical image encryption scheme using multiple chaotic maps. *Pattern Recognition Letters*, *152*, 356–364. http://dx.doi.org/10.1016/j.patrec.2021.10.033, URL: https://www.sciencedirect.com/science/article/pii/S0167865521003913.

Jasra, B., & Hassan Moon, A. (2022). Color image encryption and authentication using dynamic DNA encoding and hyper chaotic system. *Expert Systems with Applications*, *206*, Article 117861. http://dx.doi.org/10.1016/j.eswa.2022.117861, URL: https://www.sciencedirect.com/science/article/pii/S0957417422011162.

Jun, W. J., & Fun, T. S. (2021). A new image encryption algorithm based on single S-box and dynamic encryption step. *IEEE Access*, *9*, 120596–120612. http://dx.doi.org/10.1109/ACCESS.2021.3108789.

Jun, X., Xudong, Z., Xinying, X., Xiaoxia, H., Jinchang, R., & Xingbing, L. (2022). DRIN: Deep recurrent interaction network for click-through rate prediction. *Information Sciences*, *604*, 210–225. http://dx.doi.org/10.1016/j.ins.2022.04.050, URL: https://www.sciencedirect.com/science/article/pii/S0020025522004029.

Kumar, S., Kumar, M., Budhiraja, R., Das, M., & Singh, S. (2018). A secured cryptographic model using intertwining logistic map. *Procedia Computer Science*, *143*, 804–811. http://dx.doi.org/10.1016/j.procs.2018.10.386, 8th International Conference on Advances in Computing & Communications (ICACC-2018).

Lai, Q., Lai, C., Zhang, H., & Li, C. (2022). Hidden coexisting hyperchaos of new memristive neuron model and its application in image encryption. *Chaos, Solitons & Fractals*, *158*, Article 112017. http://dx.doi.org/10.1016/j.chaos.2022.112017, URL: https://www.sciencedirect.com/science/article/pii/S0960077922002272.

Li, C.-C. (2004). Asymptotic behaviors of type-2 algorithms and induced baire topologies. In *IFIP TCS*.

Li, C. (2016). Cracking a hierarchical chaotic image encryption algorithm based on permutation. *Signal Processing*, *118*, 203–210. http://dx.doi.org/10.1016/j.sigpro.2015.07.008.

Li, X., Jiang, Y., Chen, M., & Li, F. (2018). Research on iris image encryption based on deep learning. *EURASIP Journal on Image and Video Processing*, *126*, http://dx.doi.org/10.1186/s13640-018-0358-7.

Liu, Y., Cen, G., Xu, B., & Wang, X. (2022). Color image encryption based on deep learning and block embedding. *Security and Communication Networks*, *10*, http://dx.doi.org/10.1155/2022/6047349.

Ma, W., Zhou, T., Qin, J., Xiang, X., Tan, Y., & Cai, Z. (2022). A privacy-preserving content-based image retrieval method based on deep learning in cloud computing. *Expert Systems with Applications*, *203*, Article 117508. http://dx.doi.org/10.1016/j.eswa.2022.117508, URL: https://www.sciencedirect.com/science/article/pii/S0957417422008351.

Mohammed, A., Kareem, A., & Ahmed, M. (2022). Image cryptosystem for IoT devices using 2-D zaslavsky chaotic map. *International Journal of Intelligent Engineering and Systems*, *15*, 543–553. http://dx.doi.org/10.22266/ijies2022.0430.48.

Ni, R., Wang, F., Wang, J., & Hu, Y. (2021). Multi-image encryption based on compressed sensing and deep learning in optical gyrator domain. *IEEE Photonics Journal*, *13*(3), 1–16. http://dx.doi.org/10.1109/JPHOT.2021.3076480.

Özkaynak, F. (2018). Brief review on application of nonlinear dynamics in image encryption. *Nonlinear Dynamics*, *92*(2), 305–313. http://dx.doi.org/10.1007/s11071-018-4056-x.

Raghuvanshi, K. K., Kumar, S., & Kumar, S. (2020). A data encryption model based on intertwining logistic map. *Journal of Information Security and Applications*, *55*, Article 102622. http://dx.doi.org/10.1016/j.jisa.2020.102622.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, *115*(3), 211–252. http://dx.doi.org/10.1007/s11263-015-0816-y.

Setiadi, D. R. I. M., Rachmawanto, E. H., & Zulfiningrum, R. (2022). Medical image cryptosystem using dynamic josephus sequence and chaotic-hash scrambling. *Journal of King Saud University - Computer and Information Science*, *34*(9), 6818–6828. http://dx.doi.org/10.1016/j.jksuci.2022.04.002.

Sharma, S., Guleria, K., Tiwari, S., & Kumar, S. (2022). A deep learning based convolutional neural network model with VGG16 feature extractor for the detection of Alzheimer Disease using MRI scans. *Measurement: Sensors*, *24*, Article 100506. http://dx.doi.org/10.1016/j.measen.2022.100506, URL: https://www.sciencedirect.com/science/article/pii/S2665917422001404.

Sheng, Y., Li, J., Di, X., Li, X., & Xu, R. (2022). An image encryption algorithm based on complex network scrambling and multi-directional diffusion. *Entropy*, *24*(9), http://dx.doi.org/10.3390/e24091247, URL: https://www.mdpi.com/1099-4300/24/9/1247.

Suri, S., & Vijay, R. (2019). A synchronous intertwining logistic map-DNA approach for color image encryption. *Journal of Ambient Intelligence and Humanized Computing*, *10*, 2277–2290.

Teng, L., Wang, X., & Xian, Y. (2022). Image encryption algorithm based on a 2D-CLSS hyperchaotic map using simultaneous permutation and diffusion. *Information Sciences*, http://dx.doi.org/10.1016/j.ins.2022.05.032, URL: https://www.sciencedirect.com/science/article/pii/S0020025522004522.

Vaziri, M., Rahimifar, M. M., & Jahanirad, H. (2022). An enhanced chaotic system based color image encryption using dna encoding. In *2022 30th international conference on electrical engineering* (pp. 128–133). http://dx.doi.org/10.1109/ICEE55646.2022.9827353.

Wang, Y., Lei, P., Yang, H., & Cao, H. (2015). Security analysis on a color image encryption based on DNA encoding and chaos map. *Computers & Electrical Engineering*, *46*, 433–446. http://dx.doi.org/10.1016/j.compeleceng.2015.03.011, URL: https://www.sciencedirect.com/science/article/pii/S0045790615000981.

Wang, J., Long, F., & Ou, W. (2017). CNN-based color image encryption algorithm using DNA sequence operations. In *2017 international conference on security, pattern analysis, and cybernetics* (pp. 730–736). http://dx.doi.org/10.1109/SPAC.2017.8304370.

Wang, X., & Su, Y. (2021). Image encryption based on compressed sensing and DNA encoding. *Signal Processing: Image Communication*, *95*, Article 116246. http://dx.doi.org/10.1016/j.image.2021.116246, URL: https://www.sciencedirect.com/science/article/pii/S0923596521000989.

Wang, X., Su, Y., Liu, C., Li, J., Li, S., Cai, Z., & Wan, W. (2022). Security enhancement of image encryption method based on fresnel diffraction with chaotic phase. *Optical Communications*, *506*, Article 127544. http://dx.doi.org/10.1016/j.optcom.2021.127544, URL: https://www.sciencedirect.com/science/article/pii/S0030401821007938.

Wang, T., & hui Wang, M. (2020). Hyperchaotic image encryption algorithm based on bit-level permutation and DNA encoding. *Optics and Laser Technology*, *132*, Article 106355. http://dx.doi.org/10.1016/j.optlastec.2020.106355, URL: https://www.sciencedirect.com/science/article/pii/S0030399220309889.

Wang, M.-m., run Zhou, N., Li, L., & tao Xu, M. (2022). A novel image encryption scheme based on chaotic apertured fractional Mellin transform and its filter bank. *Expert Systems with Applications*, *207*, Article 118067. http://dx.doi.org/10.1016/j.eswa.2022.118067, URL: https://www.sciencedirect.com/science/article/pii/S0957417422012726.

Watson, J., & Crick, F. (1953). Molecular structure of nucleic acids; a structure for deoxyribose nucleic acid. *Nature*, *171*, 737–738. http://dx.doi.org/10.1038/171737a0.

Xu, L., Gou, X., Li, Z., & Li, J. (2017). A novel chaotic image encryption algorithm using block scrambling and dynamic index based diffusion. *Optics and Lasers in Engineering*, *91*, 41–52. http://dx.doi.org/10.1016/j.optlaseng.2016.10.012.

Yang, Y.-G., Guan, B.-W., Li, J., Li, D., Zhou, Y.-H., & Shi, W.-M. (2019). Image compression-encryption scheme based on fractional order hyper-chaotic systems combined with 2D compressed sensing and DNA encoding. *Optics and Laser Technology*, *119*, Article 105661. http://dx.doi.org/10.1016/j.optlastec.2019.105661, URL: https://www.sciencedirect.com/science/article/pii/S0030399218319030.

Ye, G., & Huang, X. (2017). An efficient symmetric image encryption algorithm based on an intertwining logistic map. *Neurocomputing*, *251*, 45–53. http://dx.doi.org/10.1016/j.neucom.2017.04.016.

Yildirim, M. (2020). DNA encoding for RGB image encryption with memristor based neuron model and chaos phenomenon. *Microelectronics Journal*, *104*, Article 104878. http://dx.doi.org/10.1016/j.mejo.2020.104878, URL: https://www.sciencedirect.com/science/article/pii/S0026269220304778.

Yildirim, M. (2022). Optical color image encryption scheme with a novel DNA encoding algorithm based on a chaotic circuit. *Chaos, Solitons & Fractals*, *155*, Article 111631. http://dx.doi.org/10.1016/j.chaos.2021.111631, URL: https://www.sciencedirect.com/science/article/pii/S0960077921009851.

Zhang, Y., Chen, A., & Chen, W. (2023). The unified image cryptography algorithm based on finite group. *Expert Systems with Applications*, *212*, Article 118655. http://dx.doi.org/10.1016/j.eswa.2022.118655, URL: https://www.sciencedirect.com/science/article/pii/S0957417422016967.

Zheng, T., Wang, Q., Shen, Y., Ma, X., & Lin, X. (2022). Batch covariance neural network for image recognition. *Image and Vision Computing*, *122*, Article 104446. http://dx.doi.org/10.1016/j.imavis.2022.104446, URL: https://www.sciencedirect.com/science/article/pii/S0262885622000750.