SAKARYA UNIVERSITY JOURNAL OF COMPUTER AND INFORMATION SCIENCES
VOL. 5, NO. 1, APRIL 2022
DOI: 10.35377/saucis.05.01.1002582

Research Article

Performance Analysis of Chaotic Neural Network and Chaotic Cat Map Based Image Encryption

Sefa Tunçer[1], Cihan Karakuzu[2]
[1]Sefa Tunçer; Bilecik Seyh Edebali University; sefa.tuncer@bilecik.edu.tr;
Bilecik Seyh Edebali University; cihan.karakuzu@bilecik.edu.tr;

Received 30 September 2021; Revised 09 February 2022; Accepted 23 February 2022; Published online 30 April 2022

Abstract

Nowadays, chaotic systems are used quite often, especially in image encryption applications. Hypersensitivity to the initial conditions, limited field-changing signs and irregular movements make chaotic systems one of the critical elements in scientific matters such as cryptography. Chaotic systems are divided in two parts as discrete time and continuous time in terms of their dimensions and properties. Gray level image encryption applications generally use one-dimensional and color image encryption applications generally use multi-dimensional chaotic systems. In this study, Tent Map, Cat Map, Lorenz, Chua, Lu chaotic systems were used for chaotic neural network based image coding application and Logistic Map and 3D Cat Map chaotic systems were used for 3D chaotic Cat Map based image encryption application. The encrypted image and the original image were examined by various analysis methods. As a result of the examinations, it is seen that both algorithms give very successful results in key size, key sensitivity, entropy analysis, histogram analysis and correlation coefficient analysis. According to the analysis, it has been shown that the chaotic neural network-based image encryption algorithm is more secure and successful.

Keywords: image encryption, encryption based on chaotic systems, chaotic neural network

1. Introduction

With the developing technology, various algorithms have been developed to ensure the security of digital images. In addition, analyzes are carried out to test the reliability of these algorithms. Use of chaotic systems in image encryption significantly increases security and efficiency in literature. In this study, performance analyzes are performed two different chaotic system based image encryption algorithms. Key size, key sensitivity, entropy analysis, histogram analysis and correlation coefficient analysis are performance criteria taken into consideration.

Prusty et al. [1] mixed the image using Arnold Cat Map chaotic system and generated key and random numbers using Henon Map and encrypted images. They have achieved successful results by encrypting images in different formats. Li et al. [2] implemented image cryptography using Tent Map and Lorenz chaotic systems. It has successfully passed key widths greater than 256 bits, randomness, histogram and correlation tests. For this reason, it can be said that there is a successful encryption. Liu and Wang [3] performed encryption and decryption by combining Red, Green and Blue (R, G, B) values in three different images of the same size, respectively. They used the SHA-256 hash function for key in encryption and Lorenz chaotic system to generate random numbers. Wang et al. [4] aimed to obtain a stronger algorithm by generating different control parameters in each iteration of image encryption. At this point, the image is encrypted with a different key at each step, and very good results are obtained in terms of speed and security. Wong et al. [5] used a chaotic Standard Map to implement an image encryption scheme based on simple addition and replacement operations. They have tried to develop this chaotic cryptographic algorithm basesd on speed, and as a result they have managed to encode a 512x512 size grayscale image below 100 milliseconds. Zeghid et al. [6] used the AES algorithm, which is used in text encryption in the literature, for image encryption. The AES algorithm was modified using a key generator to get rid of some deficiencies in the image cipher. Thus, they obtained a stronger algorithm in tests such as key width, histogram analysis, correlation coefficient and entropy analysis. Xiao et al. [7] have developed a grayscale image encryption algorithm using Arnold Cat Map and Chen chaotic systems. It has been seen to be successful based on the results obtained from the safety tests. In

this study, mathematical operations used in image encryption based on chaotic neural network (CNN) are similar to algorithm used by Xiao et al. Hongjun and Xingyuan [8] have developed a robust algorithm against noises caused by any reason using the Chebyshev Map chaotic system in the image. They obtained original image with minimum loss against the noise in the encrypted image. Randomness is increased by using MD5 in generation of encryption key to increase confidentiality. They also randomly determine the initial conditions of chaotic systems. In CNN, key is not randomly determined, but it is more secure in terms of key length.

Çavuşoğlu and Al-Sanabani encrypt 256x256 images using lightweight encryption algorithms. Small-sized images are encrypted by choosing lightweight encryption algorithms, which are generally preferred in the field of the Internet of Things. By looking at the performance analysis, it was tried to determine which encryption algorithm would be more efficient. S-AES and LBlock algorithm are suitable in terms of security and speed. Chaotic systems are more advantageous in terms of randomness and speed, but lightweight algorithms are more suitable for IoT systems [9]. Chaotic systems, which are used in many areas, are used especially in optimization algorithms to avoid local extrema and to search better in search space. Demirci and Yurtay [10] stated that chosen chaotic system can also be effective in finding global best solution in optimization algorithm. Similarly, different chaotic systems affect performance in image encryption as well. Süzen and Duman keep data on the blockchain by encrypting it with Advanced Encryption Standard (AES-256 bits) symmetric encryption algorithm. In this way, they try to ensure data integrity and confidentiality. AES, which is efficient in terms of key length and speed, is often used for text encryption [11]. Symmetric encryption algorithms do not always give desired results in histogram analysis, speed, key sensitivity and correlation coefficient analysis compared to chaotic systems in image encryption.

In the following sections, chaotic systems, chaotic neural network based color image encryption and 3-dimensions (3D) chaotic Cat Map based gray level image encryption algorithms have been investigated. The performances of these algorithms against the analysis methods are compared with each other. According to the results of the analysis, it was determined which algorithm was more effective in image encryption.

2. Chaotic Systems Used In Image Encryption

In this study, chaotic systems used for image encryption and their properties are given below. These chaotic systems differ in terms of their use and dimensions.

2.1 Tent Map

The equation of the Chaotic Tent Map system is shown in Equation 2.1 [12-13]. It is a one dimensional and discrete time system.

$$\boldsymbol{f(a,x)} = \begin{cases} \left\lceil \frac{\boldsymbol{M}}{\boldsymbol{a}} \boldsymbol{x} \right\rceil, & \boldsymbol{0 \le x \le a} \\ \left\lfloor \frac{\boldsymbol{M}}{\boldsymbol{M-a}} (\boldsymbol{M-x}) \right\rfloor + \boldsymbol{1}, & \boldsymbol{a < x \le M} \end{cases} \tag{2.1}$$

where a value $a \in [1, M]$) is an integer, and $\lfloor x \rfloor$ and $\lceil x \rceil$ represent the upper and lower limit values of x, respectively. The M value is usually chosen according to the plain text and $M = 256$ for an 8 bit image.

2.2 2D Cat Map

For an $N \times N$ gray level image, Cat Map is defined as in Equation 2.2. The p and q are the control parameters of the chaotic system and positive integers. (x, y) and (x', y') are the positions before and after coordinate values, respectively.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \boldsymbol{Q} \begin{pmatrix} x \\ y \end{pmatrix} \boldsymbol{mod}(\boldsymbol{N}) = \begin{bmatrix} \boldsymbol{1} & \boldsymbol{p} \\ \boldsymbol{q} & \boldsymbol{pq+1} \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \boldsymbol{mod}(\boldsymbol{N}) \tag{2.2}$$

2.3 3D Cat Map

This system is obtained by expanding the 2D Cat Map system. Chen et al. [14] designed to replace the pixel values of an image in a limited area without any loss (Equation 2.3). $a_x, a_y, a_z, b_x, b_y, b_z$ are control parameters that are all positive integers.

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{1 + a_x a_z b_y} & \boldsymbol{a_z} & \boldsymbol{a_y + a_x a_z + a_x a_y a_z b_y} \\ \boldsymbol{b_z + a_x b_y + a_x a_z b_y b_z} & \boldsymbol{a_z b_z + 1} & \boldsymbol{a_y a_z + a_x a_y a_z b_y b_z + a_x a_z b_z + a_x a_y b_y + a_x} \\ \boldsymbol{a_x b_x b_y + b_y} & \boldsymbol{b_x} & \boldsymbol{a_x a_y b_x b_y + a_x b_x + a_y b_y + 1} \end{bmatrix} \tag{2.3}$$

2.4 Lorenz

The Lorenz chaotic system was developed in 1962-63 (Equation 2.4). The constant parameters are $a = 10$, $b = 8/3$, $c = 28$ for the system to exhibit chaotic behavior [15-17].

$$\begin{cases} \boldsymbol{x'(t)} = \boldsymbol{a(y(t) - x(t))} \\ \boldsymbol{y'(t)} = \boldsymbol{-x(t)z(t) + cy(t)} \\ \boldsymbol{z'(t)} = \boldsymbol{x(t)y(t) - bz(t)} \end{cases} \tag{2.4}$$

2.5 Chua

The equation of the chaotic chaotic system is shown in Equation 2.5. In the system of equations, a, b are constant parameters and $f(x(t)) = 2x(t) - x(t)/7$. The system should be selected as $a = 10$, $b = 100/7$ for chaotic behavior [18].

$$\begin{cases} \boldsymbol{x'(t)} = \boldsymbol{a(y(t) - f(x(t)))} \\ \boldsymbol{y'(t)} = \boldsymbol{x(t) - y(t) + z(t)} \\ \boldsymbol{z'(t)} = \boldsymbol{-by(t)} \end{cases} \tag{2.5}$$

2.6 Lü

The equation of the chaotic chaotic system is shown in Equation 2.6. The constant parameters should be selected as $a = 36$, $b = 3$, $c = 20$ for chaotic behavior [19].

$$\begin{cases} \boldsymbol{x'(t)} = \boldsymbol{a(y(t) - x(t))} \\ \boldsymbol{y'(t)} = \boldsymbol{-x(t)z(t) + cx(t)} \\ \boldsymbol{z'(t)} = \boldsymbol{x(t)y(t) - bz(t)} \end{cases} \tag{2.6}$$

2.7 Chen

The equation of the chaotic chaotic system is shown in Equation 2.6. The constant parameters should be selected as $a = 35, b = 3$ ve $c \in [20,28.4]$ for chaotic behavior [7,20]. The change of parameter c clearly affects behavior of the chaotic system.

3. Chaotic Neural Network Based Image Encryption

A chaotic neural network (CNN) based image encryption algorithm developed by Bigdeli et al. [12] has been implemented. The encryption algorithm consists of three separate phrases consisting of a chaotic key generation block, a chaotic network layer (CNL) and a permutation network layer (PNL). The block structure of this encryption method is shown in Figure 1. The second and third layers come from 3 inputs 3 outputs and 3 neurons. The chaotic key generation block supports the network by producing appropriate weight and bias values for these layers.

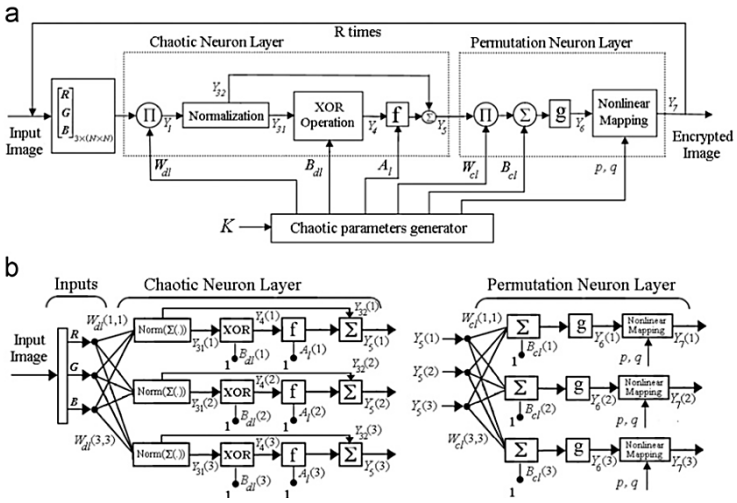

Figure 1 The encryption process (a) block scheme, (b) network scheme [12]

1. A 160-bit key is chosen for the algorithm. As shown in Fig. 2, the 160-bit key is divided into 5 groups, and $x_1(0), y_1(0), z_1(0), x_2(0), y_2(0), z_2(0), x_3(0), y_3(0), z_3(0)$ input parameters are determined. The N_0 value is chosen to avoid the problem of transiting chaotic systems. R is the number of repetitions. $x_1(N_0), y_1(N_0), z_1(N_0), x_2(N_0), y_2(N_0), z_2(N_0), x_3(N_0), y_3(N_0), z_3(N_0)$ are obtained. k is number of steps.

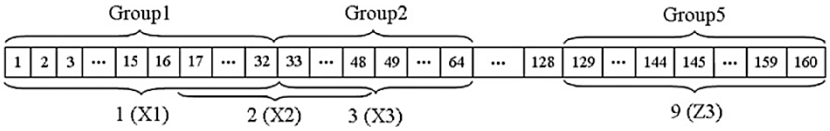

Figure 2 9 key generation from 160 bit verification code [12]

2. An F image is selected in N × N pixel dimensions. Since chaotic systems are passed N_0 times iteration, up to N_0 will not be used again. The values of the three chaotic systems are iteratively calculated for $N_0 + i$, where $i = (r-1) \times (N \times N) + 1, \dots, r \times (N \times N)$. Number of iterations of the chaotic system represents $i = 1, 2, \dots, r \times (N \times N)$. In each iteration, the values of W_{dl}, A_l and B_{dl} are calculated using Equations 3.1, 3.2, 3.3, 3.4 and 3.5, respectively.

$$\boldsymbol{W_{dl}} = \begin{bmatrix} \boldsymbol{x_1(N_0+i)} & \boldsymbol{x_2(N_0+i)} & \boldsymbol{x_3(N_0+i)} \\ \boldsymbol{y_1(N_0+i)} & \boldsymbol{y_2(N_0+i)} & \boldsymbol{y_3(N_0+i)} \\ \boldsymbol{z_1(N_0+i)} & \boldsymbol{z_2(N_0+i)} & \boldsymbol{z_3(N_0+i)} \end{bmatrix} + \boldsymbol{\alpha I} \tag{3.1}$$

$$\boldsymbol{a(j,i)} = \boldsymbol{mod}\left(\left(\left|\boldsymbol{x_j(N_0+i)}\right| - \boldsymbol{floor}\left(\boldsymbol{x_j(N_0+i)}\right)\right) \times \boldsymbol{10^{14}}, \boldsymbol{255}\right) + \boldsymbol{1}, \;\; \boldsymbol{j = 1,2,3} \tag{3.2}$$

$$\boldsymbol{A_l(i)} = [\boldsymbol{a(1,i)}, \boldsymbol{a(2,i)}, \boldsymbol{a(3,i)}]^{\boldsymbol{T}} \tag{3.3}$$

$$\boldsymbol{b(j,i)} = \boldsymbol{mod}\left(\left(\left|\boldsymbol{y_j(N_0+i)}\right| - \boldsymbol{floor}\left(\boldsymbol{y_j(N_0+i)}\right)\right) \times \boldsymbol{10^{14}}, \boldsymbol{255}\right) + \boldsymbol{1}, \;\; \boldsymbol{j = 1,2,3} \tag{3.4}$$

$$\boldsymbol{B_{dl}(i)} = [\boldsymbol{b(1,i)}, \boldsymbol{b(2,i)}, \boldsymbol{b(3,i)}]^{\boldsymbol{T}} \tag{3.5}$$

3. W_{dl} represents the weight matrix of the CNL, A_l and B_{dl} represent the bias matrices of the CNL, $mod(x, y)$ represents modulo y of x, floor (x) is less than or equal to x itself . Another matrix W_{cl}, which is the weight matrix of CNL, is used for linear mixing of the three color components obtained from the output of the chaotic neural network. It is used to change the positions of R, G, B components. Hence, it is a matrix of 3x3 dimensions, and there is only one '1' in each row and column. Equations 3.6, 3.7, 3.8 and 3.9 are used to determine W_{cl}.

$$\boldsymbol{D_i} = [\boldsymbol{x_1(N_0+i)}, \boldsymbol{y_2(N_0+i)}, \boldsymbol{z_3(N_0+i)}] \tag{3.6}$$

$$\boldsymbol{w_{1,i} = \arg\left(\max\left(D_i\right)\right)} \tag{3.7}$$

$$\boldsymbol{w_{2,i} = \arg\left(\max\left(D_i\right)\right)} \tag{3.8}$$

$$\boldsymbol{W_{cl,i}(1,w_{1,i}) = W_{cl,i}(2,w_{2,i}) = 1} \tag{3.9}$$

4. In Equations 3.7 and 3.8, arg (max (D_i)) gives index of the maximum value in the vector D_i. Next, non-zero terms of first and second rows of the matrix W_{cl} are determined. After these operations, the non-zero term of the third line is determined such that there is only one '1' in each row and column of the matrix W_{cl}. The variation of the positions of the RGB values in an image can be exemplified as in Equation 3.10.

$$\begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{1} \\ \boldsymbol{1} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{1} & \boldsymbol{0} \end{bmatrix} \times \begin{bmatrix} \boldsymbol{R} \\ \boldsymbol{G} \\ \boldsymbol{B} \end{bmatrix} = \begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{R} \\ \boldsymbol{G} \end{bmatrix} \tag{3.10}$$

5. The input image F is assumed to be N × N pixels. The size of F image is N × N × 3 with RGB. The value of k-th pixel of RGB components can be expressed as $X_k = [R_k, G_k, B_k]^T$, $k = 1, \ldots, (N \times N)$. All of the color information in the image is transformed into a matrix of three rows from $3 \times (N \times N)$. The matrix X is calculated as the input of the CNL.

   Some operations must be applied to each column of the matrix to obtain the secret information F. The operations are as follows: X_1 (Equation 3.11) is obtained based on the values F_k, $k = 1, \ldots, (N \times N)$ and $i = (r-1) \times (N \times N) + 1, \ldots, r \times (N \times N)$. Then normalization is applied (Equation 3.12) in the range of 0-255 for X_2. X_3 is obtained depending on X_2 Equation 3.13. Next, X_{31} (Equation 3.14) and X_{32} (Equation 3.15) are determined based on X_3 for use in subsequent operations. Value of X_4 (Equation 3.16) is determined by special or (XOR) operation. In this step, XOR operation means that bit XOR process. Then the chaotic activation function (Equation 3.17) is applied. The function f represents Tent Map.

$$\boldsymbol{X_{1,k} = W_{dl}(i)F(k)} \tag{3.11}$$

$$\boldsymbol{X_2(k) = Normalization(X_1(k))} \tag{3.12}$$

$$\boldsymbol{X_3(k) = floor\left(X_2(k)\right) + mod(X_2(k), floor(X_2(k)))} \tag{3.13}$$

$$\boldsymbol{X_{31}(k) = floor\left(X_2(k)\right)} \tag{3.14}$$

$$\boldsymbol{X_{32}(k) = mod(X_2(k), floor(X_2(k)))} \tag{3.15}$$

$$\boldsymbol{X_4(k) = XOR(X_{31}(k), B_{dl}(i))} \tag{3.16}$$

$$\boldsymbol{X_5(k) = f(X_4(k), A_l(i)) + X_{32}(k)} \tag{3.17}$$

6. X_5 matrix of dimensions $3 \times (N \times N)$, which is the output of the CNL, is mixed in two stages in the PNL. In the first step, each column of X_5 is mixed linearly with $X_5(k), k = 1, \ldots, (N \times N)$.

$$\boldsymbol{X_6(k) = g(W_{cl}(i)X_6(k))} \tag{3.18}$$

   The activation functions of the neurons which the parameters of the neural network layer structure, weight matrix and bias vector are arranged for determined purposes. The purpose of this study is to change the positions and pixel values of the R, G and B components using neural structure. Appropriate weight and bias values are selected for this. The result $g(x) = x$ achieves a high performance ratio, but not sometimes. The weight matrix, W_{cl}, is calculated as described in step 3.

7. In this step, the outputs of the linear permutation step are mixed. Hence, each line of the matrix X_6 is arranged in an $N \times N$ matrix so that three output $N \times N$ matrices are obtained. Then each matrix is mixed with 2B Cat Map permutation algorithm. Non-linearly mixed matrices are designated as R,

G and B values of the encrypted image, that is X_7. If the final iteration is not reached ($r < R$), $F = X_7$ is made. r is incremented by 1 and step 3 is returned. In this way, the encrypted image (F_{end}) is obtained in the last iteration and the encryption process is completed.

4. Image Encryption Based on 3d Chaotic Cat Map

3D chaotic Cat Map (CCM) based image encryption algorithm was developed by Chen et al. [14]. It comes from the following steps.

1. Key generation. 128 bit array are selected as a key and divided into a few of the parameters of the 3B Cat Map and the eight group mapped onto the Logistic Map $a_x, b_x, a_y, b_y, a_z, b_z, L_g$ and T values. The encryption block diagram is shown in Fig. 3.

2. A two-dimensional image is transformed into three-dimensional matrices. The image consists of W pixel width and H pixel height. First, all pixels of image are partitioned into a number of clusters of $N_1 \times N_1 \times N_1$, $N_2 \times N_2 \times N_2, \dots, N_k \times N_k \times N_k$ dimensions respectively. The following condition must be satisfied in order to divide an image into a matrix of several cubes.

$$\boldsymbol{W \times H = N_1^3 + N_2^3 + \cdots + N_k^3 + R} \tag{4.1}$$

where $N_k \in \{2, 3, \dots, M\}$ is the length of one edge of each cube, M is the maximum number of cubes, and $R \in \{0, 1, \dots, 7\}$ is the number of unused pixels after all cubes have been constructed.

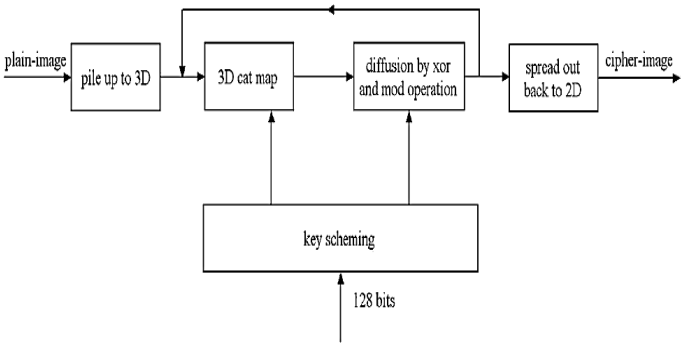

Figure 3 Image encryption block diagram of 3D Cat Map [14]

3. 3D Cat Map is performed. For each cubes mixture, a three dimensional discrete Cat Map is applied using control parameters $a_x, b_x, a_y, b_y, a_z, b_z$. In this process, the initial values $x(0) = L_g$ and $S(0) = T$ are selected and the diffusion process is performed according to the algorithm in Equations 4.2 and 4.3. L_g is the floating-point number in the range (0,1), T is an integer. L_g is the initial value of the chaotic logistic map. Eq. 5.3 expresses $E(i)$ the encrypted pixel value, $E(i-1)$the previous encrypted pixel value, $I(i)$ the current pixel value, and N the gray level image color level. In gray level images, the color level is represented by $N = 256$.

$$\boldsymbol{x(i+1) = 4x(i)[1 - x(i)]} \tag{4.2}$$

$$\boldsymbol{E(i) = \varphi(i) \oplus \{[I(i) + \varphi(i)\,]modN\} \oplus E(i-1)} \tag{4.3}$$

4. The cubes mixed in three dimensions are transformed into two dimensional images again.

5. The operations are repeated regularly until security reaches a appropriate level in steps 3 and 4. As the number of iterations increases, a more secure encryption takes place, but disadvantages arise, such as computational cost and time delays.

s

5. Analyzes and Conclusions

The security levels of CNN and CCM based image encryption algorithms have been tested. Various analyzes have been made for this. Analyzes have shown that the CNN is safer.

5.1 Key size security

The key size ensures that the encryption algorithm is robust against brute force attacks. The CNN has 224 bits and the 3B CCM has a 128 bit long key. From this point of view, CNN is more secure against brute force attacks.

5.2 Key sensitivity

Chaotic systems considerably increase key sensitivity. When the input parameters of the chaotic system obtained by the key change 10^{-14}, it becomes impossible to obtain the original image from the encrypted image. This sensitivity can be up to 10^{-16} at some points. From this point of view, CNN system, which uses more chaotic systems, is more secure.

5.3 Information entropy analysis

The degree of uncertainty in a system is entropy. The information entropy $E(m)$ of an m message is calculated as in Equation 5.1 [12].

$$E(m) = -\sum_{j=0}^{2^n-1} p(m_j) log_2 \frac{1}{p(m_j)} \tag{5.1}$$

$p(m_j)$ is the probability of occurrence of m_j. Each symbol has equal probability $p(m_j) = 2^{-8}$, if $n = 8$. In this case, the distribution term in the entropy is $E(m) = 8$. The entropy values of the RGB in the encrypted image are given in Table 1. Both CNN and CCM have successful results.

Table 1 Information entropy analysis values

Method	**CNN**			**CCM**
	R	G	B	Gray Level
Lena	7.9928	7.9937	7.9928	7.9967
Baboon	7.9931	7.9938	7.9931	7.9970
Peppers	7.9930	7.9934	7.9929	7.9971

5.4 Histogram analysis

The histograms of the images encoded with CNN and CCM are shown in Fig. 4 and Fig. 5, respectively. It shows an almost homogeneous distribution, when the histogram of encrypted images is examined. At this point, no information about the pixel in the original image can be obtained from the encrypted image. Therefore, there is no statistical attack on the encrypted image, the original image and the proposed encryption process. Both CNN and CCM showed successful results in histogram analysis.

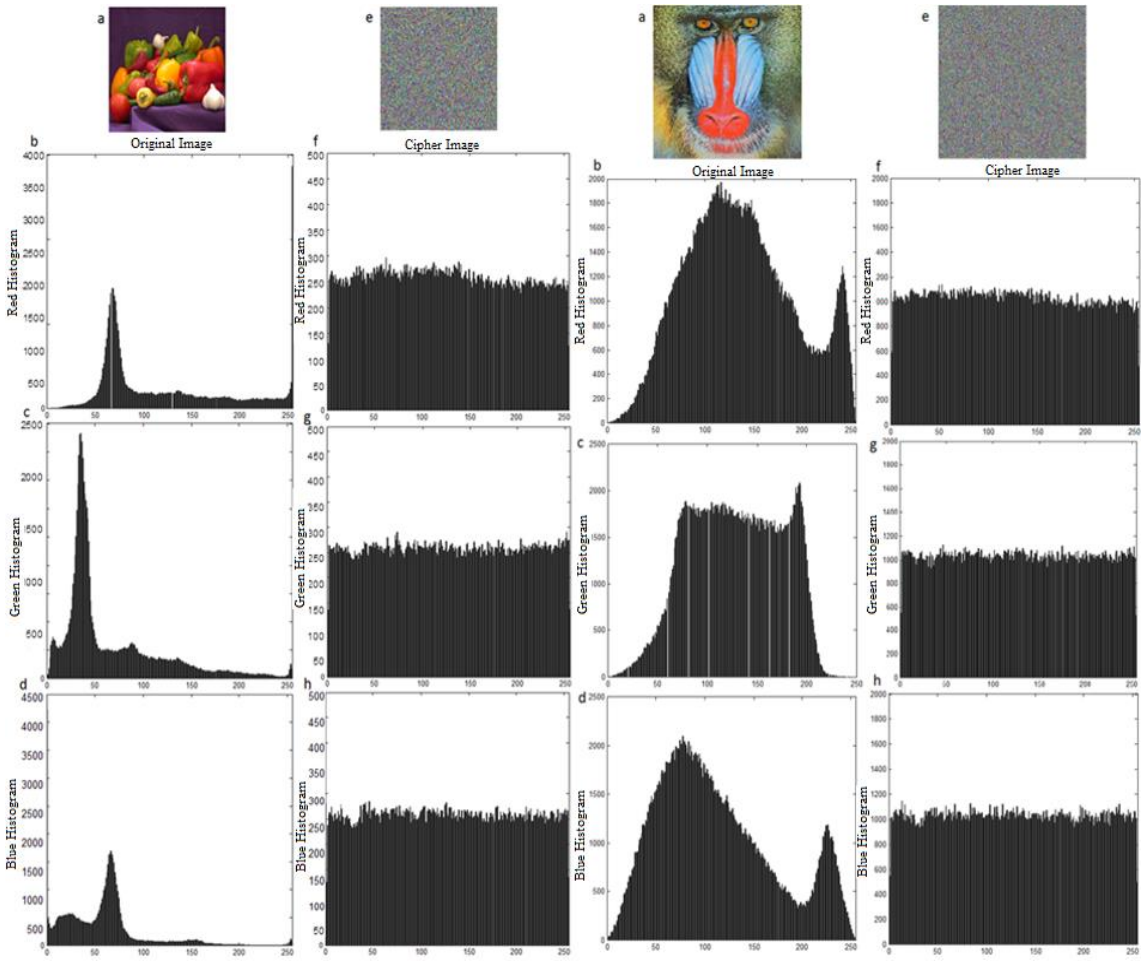

Figure 4 Red (b, f), green (c, g) and blue (d, h) color histograms of the original (a) and encrypted (e) states of peppers and baboon images for CNN

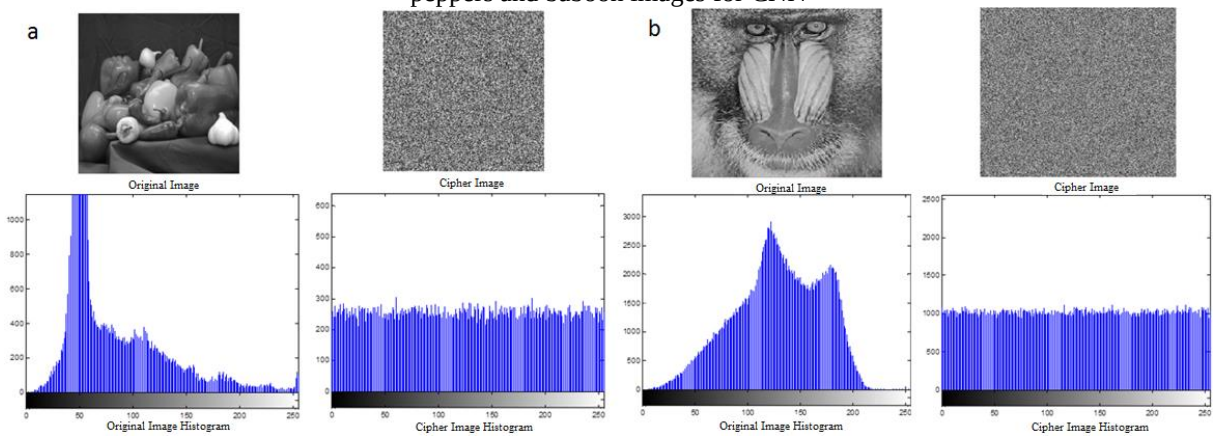

Figure 1 Gray level histograms of original and scrambled states of peppers (a) and baboon (b) images for CCM

5.4 Correlation coefficient analysis

Correlation explores the relationship between two or more values in terms of decrease and increase. Correlation value in an original image will be high. An effective image encryption algorithm should reduce the correlation between adjacent pixels. The average correlation values of the vertical, horizontal and diagonal pixels of 3 different images encrypted with CNN and CCM are shown in Table 2. According to these results, it can be said that CNN is more successful.

Table 2 Vertical, horizontal and diagonal correlation values

	Original Image			**Cipher Image (CNN)**			**Cipher Image (CCM)**		
	Vertical	Horizontal	Cross	Vertical	Horizontal	Cross	Vertical	Horizontal	Cross
Lena	0.94	0.9227	0.8983	-0.0432	-0.0145	-0.0011	-0.2035	0.0008	-0.007
Baboon	0.7585	0.83	0.7727	0.0079	-0.0038	-0.0273	-0.3078	-0.0031	0.003
Peppers	0.9772	0.9793	0.9630	0.0098	-0.0141	0.0143	-0.0841	-0.0100	-0.011

5000 adjacent pixel pairs were randomly selected from the baboon image for CNN. The vertically, horizontally and diagonally adjacent pixels in the original and ciphered images are compared (Figure 6). It is seen that the correlations of the adjacent pixels of the original image are highly close to each other and the adjacent pixels of the encrypted image is close to zero.

Similar to CNN, 5000 adjacent pixel pairs were randomly selected from the baboon image for CCM. The vertically, diagonally adjacent pixels in the original and ciphered images are compared (Figure 7). It is seen that the correlations of the adjacent pixels of the original image are highly close to each other and the adjacent pixels of the encrypted image is close to zero.

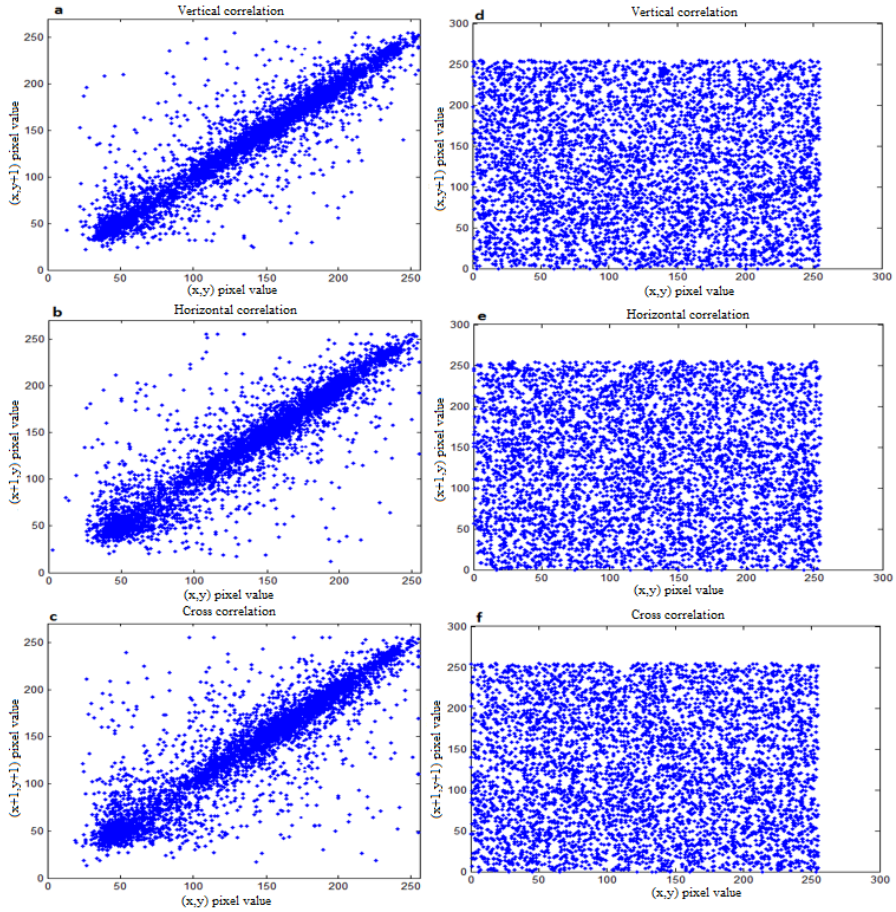

Figure 2 Vertical, diagonal correlation graphs of original (a, b, c) and ecnrypted (d, e, f) baboon images for CNN

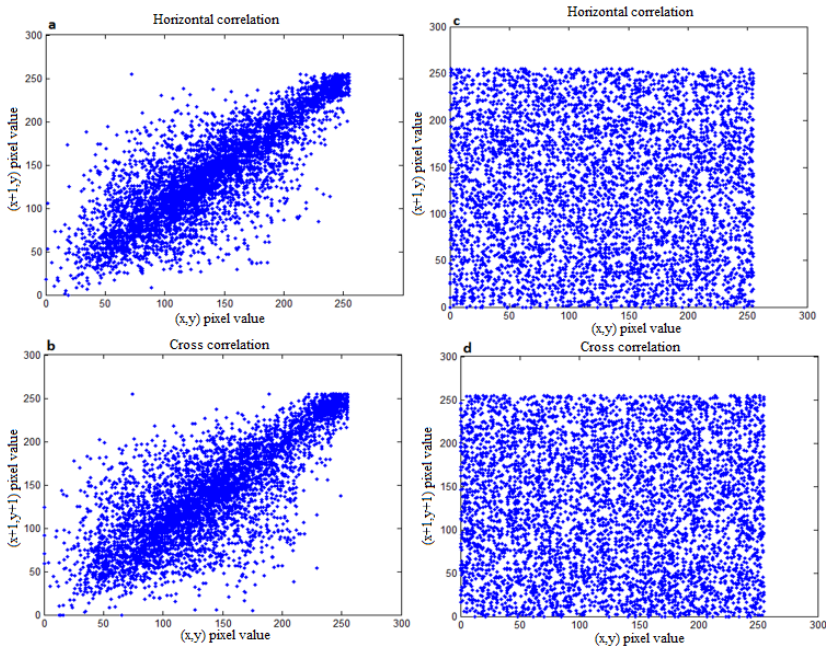

Figure 3 Vertical, horizontal, diagonal correlation graphs of original (a, b) and encrypted (c, d) baboon images for CCM.

Based on all analyzes, CNN and CCM algorithms can be simply compared as in Table 3. Key sensitivity line looks same for both algorithms, it has a more precise and effective structure due to the chaotic systems used by the CNN. While the correlation coefficients are at the maximum level of 0.01 in CNN, this ratio can be at level of 0.3 in CCM depending on selected picture. Therefore, it can be said that CNN is better in terms of correlation analysis. As a result, both algorithms gave desired results, but CNN was found to be a safer and more successful algorithm than CCM. In addition, both gray level and color image encryption can be done with CNN.

Table 3 Analysis results of CNN and CCM based algorithms

Parameter / Method		**CNN**	**CCM**
Key size security		224 bit	128 bit
Key sensitivity		10^{-14} degrees	10^{-14} degrees
Histogram analysis		Secure	Secure
Correlation coefficient analysis	Vertical	-0.0432	-0.2035
	Horizontal	-0.0145	-0.0008
	Cross	-0.0011	-0.0070
Information entropy analysis		R: 7.9930 G: 7.9938 B: 7.9929	Gray: 7.9921

References

[1] A.K. Prusty, A. Pattanaik, S. Mishra, “An Image Encryption & Decryption Approach Based on Pixel Shuffling Using Arnold Cat Map & Henon Map”, International Conference on Advanced Computing and Communication Systems, 1-6, 2013.

[2] J. Li, Y. Xing, C. Qu, J. Zhang, “An Image Encryption Method Based on Tent and Lorenz Chaotic Systems”, Software Engineering and Service Science (ICSESS), 582-586, 2015.

[3] H. Liu and X. Wang, “Triple-image encryption scheme based on one-time key stream generated by chaos and plain images”, The Journal of Systems and Software, 86:826-834, 2013.

[4] Y. Wang, K. Wong, X. Liao, T. Xiang, G. Chen, “A chaos-based image encryption algorithm with variable control parameters”, Chaos, Solitons and Fractals, 41:1773-1783, 2009.

[5] K. Wong, B.S. Kwok, W.S. Law, “A fast image encryption scheme based on chaotic standard map”, Elsevier Physics Letter A, 372(15):2645-2652, 2008.

[6] M. Zeghid, M. Machhout, L. Khriji, A. Baganne, R. Tourki, "A Modified AES Based Algorithm for Image Encryption", International Journal of Computer Science and Engineering, 1(1):70-75, 2007.
[7] D. Xiao, X. Liao, P. Wei, "Analysis and improvement of a chaos-based image encryption Algorithm", Chaos, Solitons and Fractals, 40:2191–2199, 2009.
[8] L. Hongjun and W. Xingyuan, "Color image encryption based on one-time keys and robust chaotic maps", Computers and Mathematics with Applications, 59:3320-3327, 2010.
[9] Ü. Çavuşoğlu and H. Al-Sanabani, "The Performance Comparison of Lightweight Encryption Algorithms", Sakarya University Journal of Computer And Information Sciences, 2(3):158-169, December 2019.
[10] H. Demirci and N. Yurtay, "Effect of the Chaotic Crossover Operator on Breeding Swarms Algorithm", Sakarya University Journal of Computer And Information Sciences, 4(1):120-130, April 2021.
[11] A. A. Süzen and B. Duman, "Blockchain-Based Secure Credit Card Storage System for E-Commerce", Sakarya University Journal of Computer And Information Sciences, 4(2):204-215, August 2021.
[12] N. Bigdeli, Y. Farid, K. Afshar, "A novel image encryption/decryption scheme based on chaotic neural networks", Engineering Applications of Artificial Intelligence 25:753–765, 2012.
[13] N. Masuda and K. Aihara, "Cryptosystems With Discretized Chaotic Maps", IEEE Transactions On Circuits And Systems: Fundamental Theory And Applications, 49(1):28-40, 2002.
[14] G. Chen, Y. Mao, C.K. Chui, "A symmetric image encryption scheme based on 3D chaotic cat maps", Elsevier Chaos, Solitons and Fractals, (21):749–761, 2004.
[15] E.N. Lorenz, "Deterministic Nonperiodic Flow", Journal of the Athmosferic Sciences, 20:130-141, 1963.
[16] O.A. González, G. Han, J.P. de Gyvez, and Edgar, "CMOS Cryptosystem Using a Lorenz Chaotic Oscillator", Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS '99, 5:442-445, 1999.
[17] D. Li, Z. Yin, "Connecting the Lorenz and Chen systems via nonlinear control", Commun. Nonlinear Sci. Numerical Simulation, 14(3):655–667, 2009.
[18] T. Botmart and P. Niamsup, "Adaptive control and synchronization of the perturbed Chua's system", Math. Comput. Simulation, 75(1–2):37–55, 2007.
[19] J. Lü, G. Chen, S. Zhang, "The compound structure of a new chaotic attractor" Chaos Solitons Fractals, 14(5):669–672, 2002.
[20] G. Chen and T. Ueta, "Yet another chaotic attractor", Int J Bifurcat Chaos, 9(7):1465–6, 1999.