TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

THESIS NO: 072-MSCSKE-665

SECURITY ENHANCEMENT OF AN IMAGE ENCRYPTION SCHEME BASED ON CHAOTICALLY COUPLED CHAOTIC MAPS

BY

SAROJ THAPA

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF COMPUTER SYSTEM AND KNOWLEDGE ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

November, 2018

Security Enhancement of an Image Encryption Scheme Based On Chaotically Coupled Chaotic Maps

By

SAROJ THAPA

072-MSCSKE-665

Supervised by:

Dr. Dibakar Raj Pant

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer system and knowledge Engineering

Department of Electronics and Computer Engineering
Institute of Engineering, Pulchowk Campus
Tribhuvan University
Lalitpur, Nepal

November , 2018

COPYRIGHT©

TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

The undersigned certify that they have read, and recommended to the Institute of Engineering for acceptance, a thesis entitled **"Security Enhancement of an Image Encryption Scheme Based on Chaotically Coupled Chaotic Maps"** submitted by **Saroj Thapa [072-MSCSKE-665]** in partial fulfillment of the requirements for the degree of **Master of Science in Computer System and Knowledge Engineering**.

Supervisor,
Dr. Dibakar Raj Pant
Associate Professor
Department of Electronics and Computer Engineering
Pulchowk Campus

External Examiner,

Dr. Pradip Paudyal

Deputy Director

Nepal Telecommunications Authority

Committee Chairperson,

Dr. Aman Shakya

Program Co-ordinator

Computer System and Knowledge Engineering

Date: 26 November, 2018

DEPARTMENTAL ACCEPTANCE

The thesis entitled **"Security Enhancement of an Image Encryption Scheme Based on Chaotically Coupled Chaotic Maps"** submitted by **Saroj Thapa (072-MSCSKE-665)** for the partial fulfillment of the requirements for the award of the degree of **"Master of Science in Computer System and Knowledge Engineering"** has been accepted as a bonafied record of work independently carried out by him in the department.

Dr. Surendra Shrestha
Head of the Department
Department of Electronics
and Computer Engineering
Pulchowk Campus,
Tribhuvan University,
Nepal

ACKNOWLEDGEMENT

A big thanks and profound gratitude goes to my supervisor **Dr. Dibakar Raj Pant**, for his essential and precious guidelines. It has been a privilege to work under his supervision.

I would like to express my special thanks of gratitude to the Department of Electronics and Computer Engineering (DOECE) and to our Head of Department **Dr. Surendra Shrestha** for providing us with the golden opportunity to explore our interest and ideas in the field of engineering through this thesis.

I would like to express my sincere gratefulness to the Department of Electronics and Computer Engineering and MSCSKE Program Coordinator, **Dr. Aman Shakya**, for providing me with the opportunity.

I would like also like to thank **Prof. Dr. Shashidhar Ram Joshi** for providing me necessary suggestions for the inception of this research work. I would also like to express gratitude to **Prof. Dr. Subarna Shakya, Dr. Sanjeeb Panday** and other faculty members of the Department for their kind assistance and guidance.

I would like to thank my family and friends for providing me moral support, helping me get all the resources required for the accomplishment of this thesis.

ABSTRACT

This research aims to enhance the security of image encryption scheme using the application of chaos theory. Due to some intrinsic features of the images, such as bulk data capacity and high correlation among pixels, the traditional encryption techniques such as AES, DES, RSA, etc. are not suitable for image cryptography. So chaos based image cryptography is introduced to overcome these drawbacks associated with traditional encryption techniques. This image encryption scheme encompasses confusion and diffusion stages. During confusion stage, three levels of shuffling using different chaotic maps are used. In the first level, cubic map is used and Arnold cat map in the second level followed by tent map. Finally the shuffled image is diffused using Henon map to produce the ciphered image for transmission. Sensitivity analysis is performed with respect to key to its security against brute-force attacks. The average entropy value of 7.97 provides vigorous upon entropy attack. The Number of Pixel Change Ratio (NPCR) of above 99% obtained from the diffusion stage signify the use of multiple chaotic maps. The image cryptosystem has 2^{128} different combinations of the secret keys. The maximum correlation values in vertical, horizontal and diagonal directions of encrypted images are 0.0826, 0.0772 and 0.0721 respectively and structured similarity index measurement of not more than 0.0152 between original image and encrypted image validates the robustness of this algorithm.

Keywords: Chaos Theory, tent map, Cubic map, Arnold cat map, Tent map, Henon map

Table of Contents

Tables of Figures

List of Tables

List of Abbreviation

AES	Advanced Encryption Standard
DES	Data Encryption Standard
MSE	Mean Square Error
NPCR	Number of Pixel Change Rate
PSNR	Peak Signal to Noise Ratio
PRKG	Pseudo Random Key Generator
RSA	Rivest-Shamir-Adleman
SSIM	Structured Similarity Index measurement

Chapter One: Introduction

1.1 Introduction

1.1.1 Overview

Images are widely used in daily life, and as a result, the security of image data is an important requirement. Encryption is performed when it is necessary to protect user privacy. Image-encryption algorithms are used to provide this security and can be divided into two groups with respect to the approach used to construct the encryption scheme**:** chaos-based methods and non-chaos-based methods [1].

The implemented method of image cryptosystem is based on chaos theory. This theory concerns deterministic systems whose behavior can be predicted. These systems can be predicted for a while and then they become random. This research work is based on implementation of multiple chaotic maps to encrypt and decrypt an image. The plain-image is first divided into blocks and then performs three levels of shuffling using different chaotic maps. In the first level the pixels within the block are shuffled using cubic map [2]. In the second level the blocks are shuffled using Arnold cat map [3] and all the pixels in an image are shuffled using tent map [4] in the third level. Finally, the shuffled image is diffused using a chaotic sequence generated by using Henon map [5] to produce the ciphered image for transmission.

The security and statistical analysis including sensitivity analysis, statistical analysis, entropy testing, correlation analysis as well as structural similarity index measure of image encryption scheme ensure the robustness and efficiency of implemented algorithm. Through these security parameters, one can determine a better and highly secure image encryption scheme for transmission.

1.1.2 Basic Concept on Chaos Theory

The term chaotic comes from chaos. Chaos does not have a defined meaning; it may refer to a state that does not have deterministic behavior. Chaotic systems depend completely on initial condition. The butterfly effect [1] is the concept that small causes can have large effects. Initially, it was used with weather prediction but later the term became a metaphor used in and out of science. These systems are dynamic therefore with a chaotic system the results vary largely with a little change in initial condition. Chaotic theory is a field of mathematics and has various applications in meteorology, economics, philosophy etc. Various image encryption techniques have been implemented during the last years based on multiple one-dimensional, two-dimensional or higher-dimensional chaotic systems, coupled chaotic maps etc. Chaos theory concerns deterministic systems whose behavior can be predicted. These systems can be predicted for a while and then they become random. Lorenz[3] discovered the effect when he observed that runs of his weather model with initial condition data that was rounded in a seemingly inconsequential manner would fail to reproduce the results of runs with the unrounded initial condition data. Chaos-Based techniques have been extensively studied in the recent years, because their properties lead to the potential cryptography. Confidentiality, non-periodicity, more randomness and easy implementation are the main advantages of using chaos theory in image encryption.

Chaos is a promising candidate for cryptography. The highly sensitive behavior of chaotic maps to the initial conditions is directly connected to the two basic properties of good cipher: confusion and diffusion. Chaos is a ubiquitous phenomenon existing in deterministic nonlinear systems that exhibit extreme sensitivity to initial conditions and have random like behavior. Chaos theory is based on the observation that simple rules when iterated can give rise to apparently complex behavior [3].

1.2 Problem Statement

Image encryption can be implemented by altered methods of encryption such as Rivest-Shamir-Adleman (RSA), advanced encryption standard (AES) and data encryption standard (DES) [6]. Some of which are insufficient to meet protection. These techniques are simply for attacks. Image Encryption is different than text encryption due to bulk data capacity, high correlation among pixels and high Redundancy, these intrinsic features are difficult to handle by traditional ciphers. As a result, cryptographic techniques are required to accomplish a certain level of security, integrity, confidentiality and as well as, to prevent unauthorized access of sensitive information during data storage and transmission of an image. For these causes to guarantee the security, image encryption using various chaotic maps system is introduced.

1.3 Objective

Following are the major objectives of this research:

- To encrypt and decrypt images using different chaotic sequences generated from various chaotic maps.
- To evaluate the performance of image cryptosystem using key sensitivity analysis, entropy analysis, Number of Pixel Changing Ratio, Correlation analysis and Structural Similarity Index Measurement.

Chapter Two: Literature Review

2.1 Related Works

This section presents the research work of some prominent authors in the same field and explaining a short description of various chaos based techniques used for image encryption.

Chenghang Yu, et al. [1] analyzed the chaotic features of trigonometric function and implemented a new algorithm based on the trigonometric function for fast and secure image encryption. Large quantity of experiment data and performance analysis prove that the trigonometric function is of excellent chaotic features and is very suitable for image encryption. Trigonometric function is one of the most basic and important function in nature. In fact, not all of the trigonometric functions can be used for encryption. The encryption feature of a trigonometric function is determined by the parameters such as the frequency and the phase. The use of trigonometric function does not provide sufficient keyspace for image encryption.

C.K. Huang and H.H. Nien [2] introduces a new pixel shuffle technique with multi chaotic systems for the image encryption. Correlation coefficient, NPCR, and UACI has been conducted to test on the security analysis and the distribution of distinguished elements of variables for the encrypted image. The adopted examples show the confidential encrypted images and demonstrate a good potential in the application of the digital-color image encryption. It only employs two chaotic maps for image encryption.

Shubo Liu, et al. [3] offered another new encryption algorithm by investigating the principle of the algorithm of chaos encryption focused around logistic map. Furthermore, the security and achievement of the projected algorithm is also appraised. and the coupled chaotic maps demonstrates benefits of enormous key space and advanced security. The system is formed in a stream-cipher structural design, where the PRKG is made by two chaotic maps, allocating the resolution of stream generation and random mixing, correspondingly. It is observed that such a design can improve the randomness, but below finite precision implementation.

Rajinder Kaur and Er.Kanwalprit Singh [6] worked based on the chaotic encryption and Improved DES encryption and a combination of image encryption algorithm is used to find the gaps. In this paper new encryption logistic Map produced pseudo random sequence on RGB image and make double times encryption with improved DES. But the with the use of DES if the message is encrypted with a particular key, and is taken 1's compliment of that, encryption will be same as that of the encryption of the compliment message and compliment key.

Komal D Patel and Sonal Belani [7] implemented new image encryption technique based on new chaotic system by adding two chaotic systems: the Lorenz chaotic system and the Rössler chaotic system. From Experimental analysis they demonstrate that the image encryption algorithm has the advantages of large key space and high-level security but only employs two chaotic maps for encryption.

Rashidah Kadir, et al. [8] implemented image encryption scheme an external secret. For image encryption two chaotic logistic maps are employed. The initial conditions for the both logistic maps are derived using the external secret key by providing different weight age to its bits. In the algorithm, the first logistic map is used to generate numbers ranging from 1 to 24 (numbers may be repeated). The initial condition of the second logistic map is modified from the numbers, generated by the first logistic map.

XiaoJun Tong, et al. [9] implemented encryption algorithm includes two parts: firstly, the positions of the original image pixels are permuted by Baker map**;** secondly, the values of the permuted pixels are encrypted by multiple chaotic map. Baker map is used to permute the positions of image pixels in the spatial-domain At the same time, the probability of precision degradation is lower than simple-chaotic map encryption scheme.

Hooman Kashanian, et al. [10] presented a novel scheme for image encryption. At first, a two dimensional logistic mapping is applied to permutation relations between image pixels. fractal image as an encryption key has been used. Given that the chaotic mapping properties such as extreme sensitivity to initial values, random behavior, non-periodic, certainty and so on, theses mappings are used in order to select fractal key for encryption.

Chapter Three: Research Methodology

3.1 System Architecture

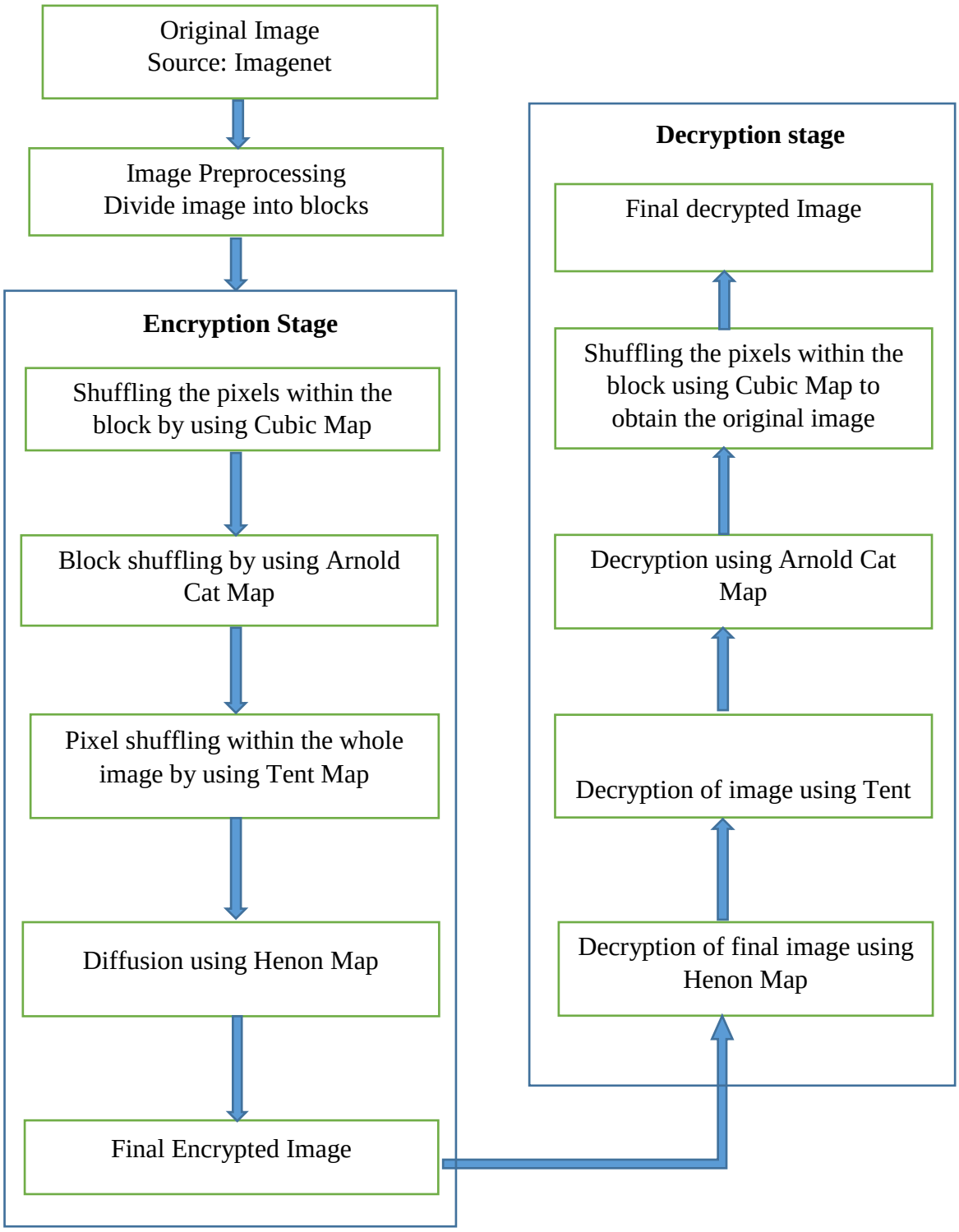


Figure 3.1 System Architecture for chaos based image cryptography

3.2 Dataset

The ***image-net*** dataset have been used for the implementation of this algorithm. This is huge image dataset consisting of 14,197,122 images comprising different formats (.jpg,.jpeg,.png,.tif,.bmp).

3.3 Confusion stage

The confusion stage is the pixel permutation where the position of the pixels is scrambled over the entire image without disturbing the value of the pixels and the image becomes unrecognizable. Three levels of shuffling with different chaotic maps has been performed in this stage. The steps to be followed in this confusion stage are:

Step1: An image of size NxN has been divided into 10x10 sized blocks.

Step2: The pixels within the block has been shuffled using Cubic Map.

Step3: All the 10x10 blocks within an image has been shuffled using Arnold Cat map.

Step4: The pixels in the whole image has been shuffled using Tent Map

3.3.1 Image Encryption using Cubic map

The Cubic map [4] is a discrete-time dynamical system. It is an example of a dynamical system that exhibit chaotic behavior .Here, the one-dimensional map is mapped into a ternary string via symbolic dynamics in order to evaluate the complexity. The cubic map equation is given by equation [1].

$$x_{t+1} = rx_t^3 + (1 - r)\, x_t \quad (1)$$

Here, x_t is the initial parameter to generate the chaotic sequence. The map depends on the value r which is called as bifurcation parameter. This will be usually of degree 3 to produce chaotic behavior.

.

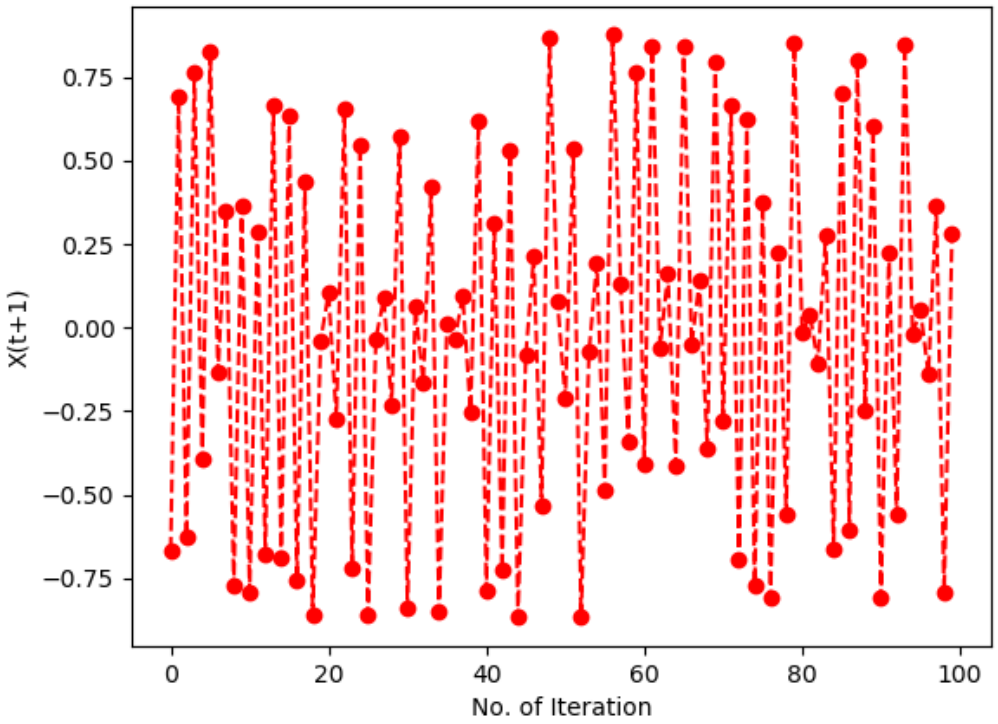


Figure 3.2 Analysis of chaotic behavior of cubic map with initial key (x=0.678) and bifurcation parameter (r=3.67)

For the value of bifurcation parameter (r=3.670) and initial condition(x=0.678) chaotic behavior of cubic map can be observed for 100 iterations. The slight variations in the initial condition yield dramatically different results over time and it a prime characteristic of any chaotic system.

3.3.1.1 Flow chart

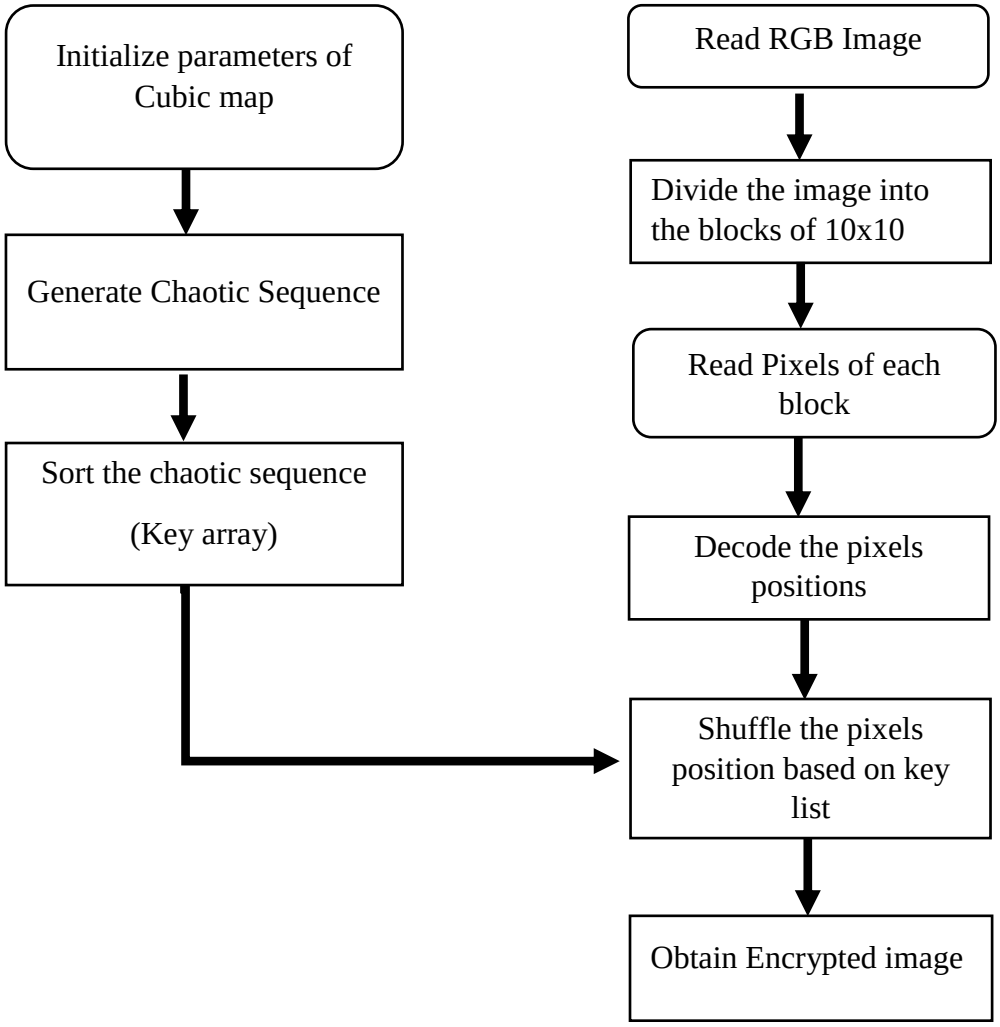


Figure 3.3 Flowchart of Image encryption by using cubic map

3.3.3 Image encryption using Arnold Cat Map

Arnold's Cat Map are chaotic two dimensions that can be used to change the position of the pixel of the image without removing any information from the image. This Map is a transformation that can be applied to an image. The pixels of the image appear to be randomly rearranged, but when the transformation is repeated enough times, the original image will reappear [6].

ACM transform the coordinates (x, y) in the image of size N × N to the new coordinates (x_{i+1}, y_{i+1}). The equation of ACM is given in equation [2].

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} mod(N) \quad (2)$$

Where (xi, yi) is position of the pixel in the image, (x_{i+1}, y_{i+1}) is new pixel position after iteration i. Parameters a and b are any positive integer. ACM is iterated as N times and each iteration produces a random image. Values of a, b and N can be considered as the secret keys. The scramble image can be reconstructed into the original image using the same key (a, b and N). The inverse equation of ACM is shown in equation [3].

$$\begin{bmatrix} x_i \\ yi \end{bmatrix} = \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix}^{-1} \begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} mod(N) \quad (3)$$

3.3.3.1 Flowchart

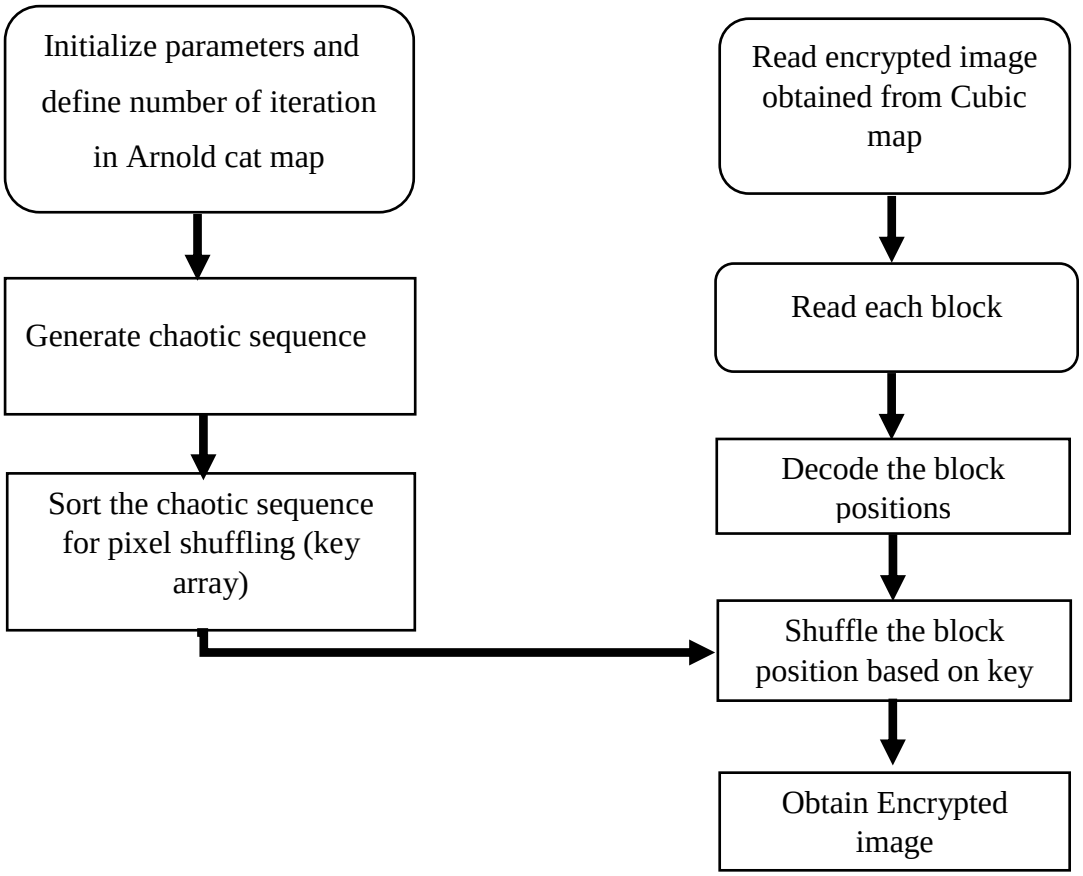


Figure 3.4 Flow chart of Image encryption by using Arnold Cat Map

3.3.4 Image encryption using Tent Map

The tent map[4] is a piecewise linear, one-dimensional map . Like the logistic map, its iterates exhibit chaotic dynamics for a range of values of the parameter.It is exhibited by following equation [4].

$$X(t+1)=\begin{cases}\mu x_t & 0 \le x_t \le \frac{1}{2} \\ \mu x_t(1-x_t), & 1 \ge x_t \ge \frac{1}{2}\end{cases} \quad (4)$$

where $x_t \in [0, 1]$.. This map transforms an interval [0, 1] onto itself and contains only one control parameter μ, respectively, where $\mu \in [0, 2]$. The tent map is topologically conjugate, and thus the behaviors of the map are in this sense identical under iteration. Although the form of the tent map is simple and the equation involved is linear, for certain parameter values, this system can display highly complex behavior and even chaotic phenomena[4].

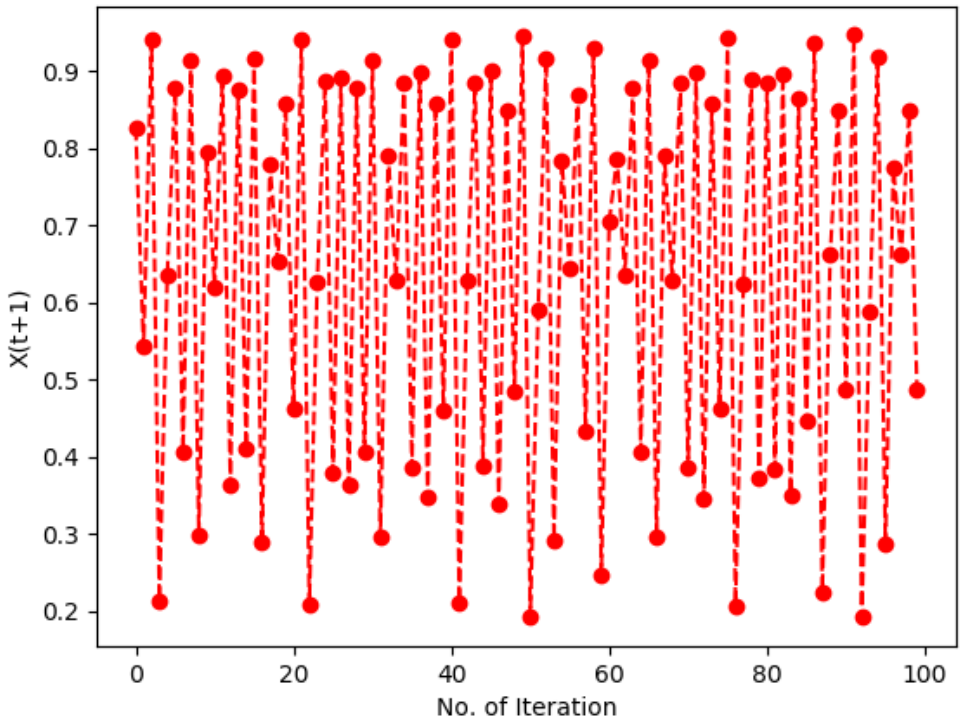


Figure 3.5 Analysis of chaotic behavior of tent map with initial key (x_t=0.678 and bifurcation parameter (r=1.7876)

3.3.4.1 Flow chart

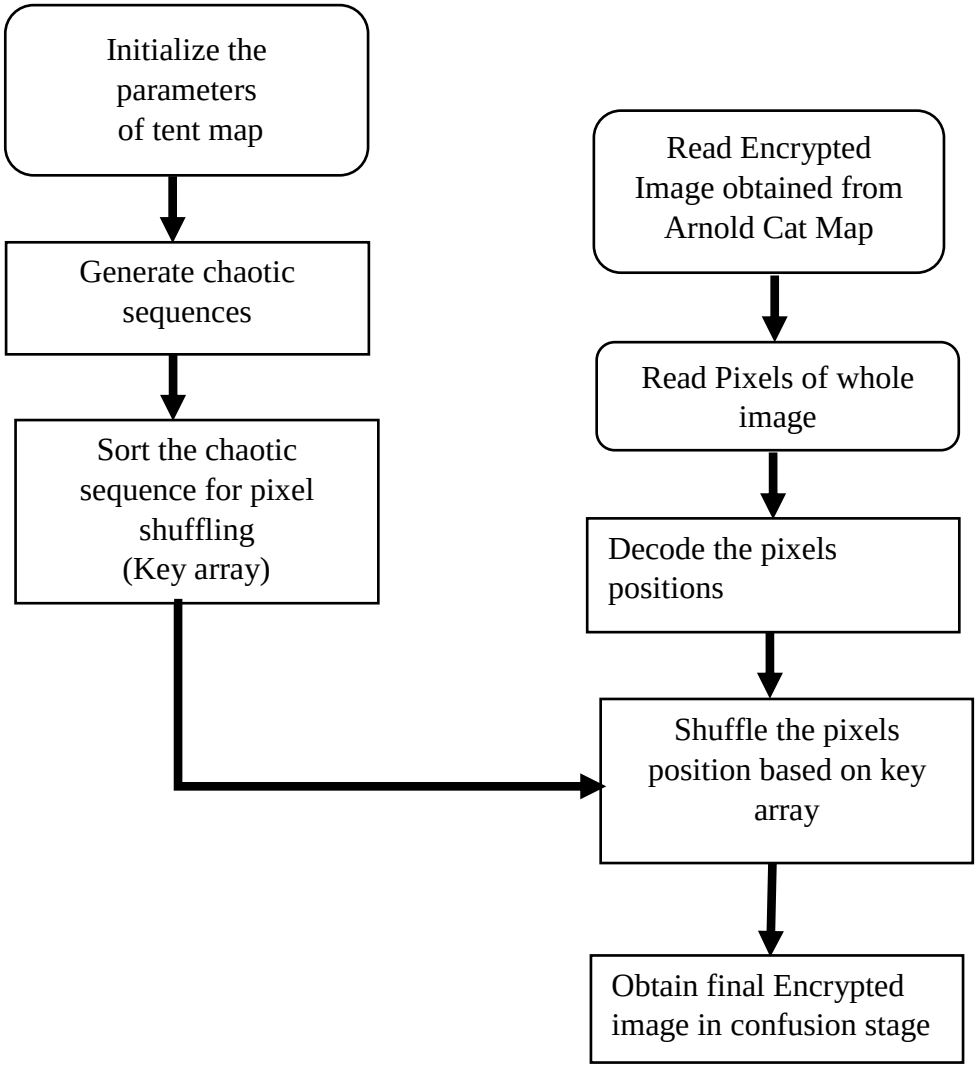


Figure 3.6 Flow chart of Image encryption using Tent map

3.4 Pixel shuffling in confusion stage

This pixel shuffling algorithm is based on pixel scrambling where in the randomness of the chaos is made utilized to scramble the position of the data. The position of the data is scrambled in the order of randomness of the elements obtained from the chaotic map and again rearranged back to their original position in decryption process.

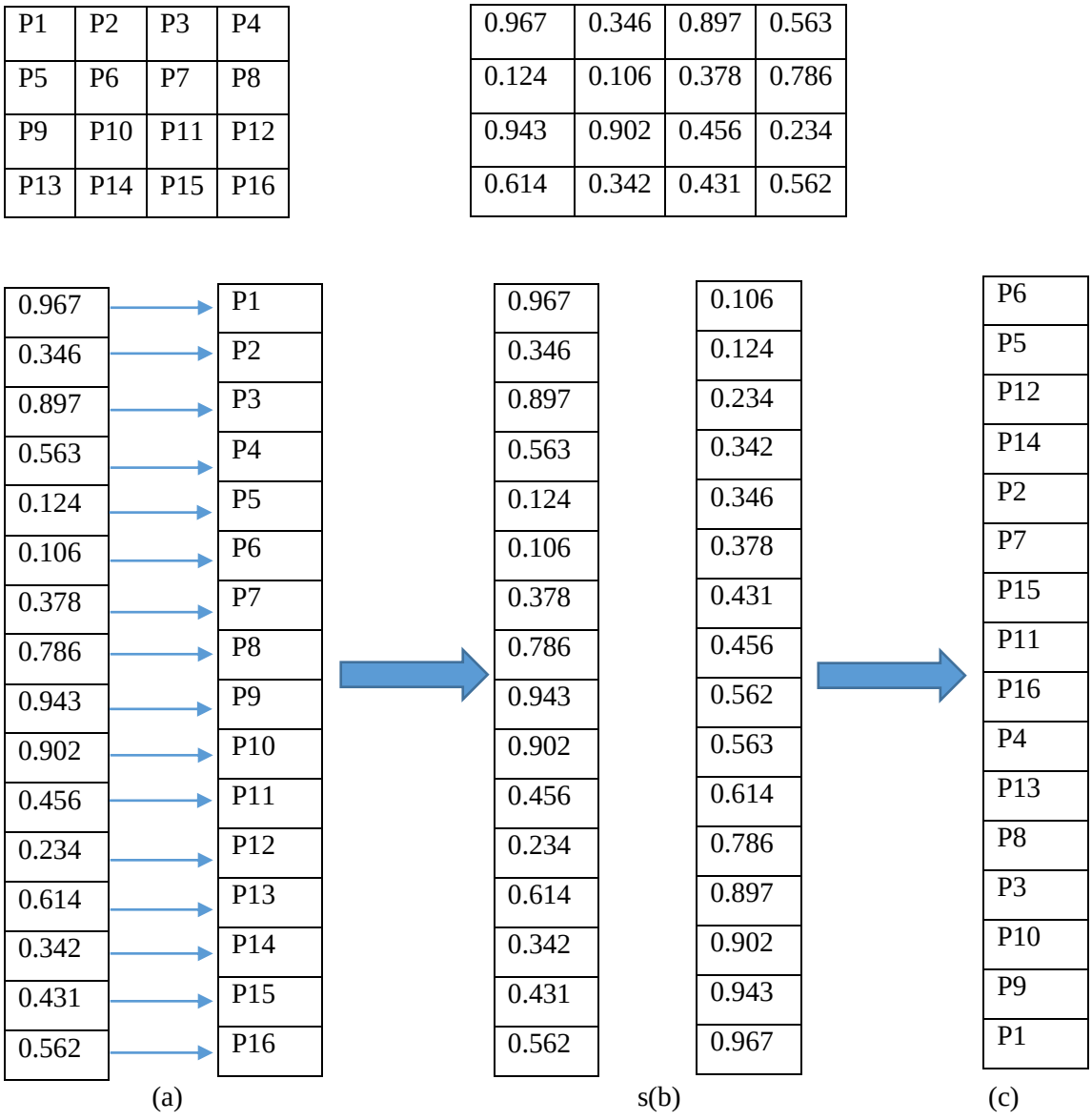


Figure 3.7 a. Generation of chaotic sequence equals to number of pixels b. Reordering the chaotic sequence generated into ascending order c. shuffling the image pixels on the notion of change in the position of chaotic sequence.

3.5 Diffusion Stage

In the diffusion stage, the pixel values has been modified sequentially by the sequence generated by the Henon map. After confusion stage even though the pixels are shuffled the histogram remains unchanged. Here pixel values has been modified to get a normalized histogram. Diffusion has been performed using XOR operation.

3.5.1 Image Encryption using Henon Map

The Henon map is a discrete-time dynamical system. It is one of the most studied examples of dynamical systems that exhibit chaotic behavior. The Henon map takes a point (x_i, y_i) in the plane and maps it to a new point given by the equation [6].

$$x_{i+1} = 1 - ax_i^2 + y_i \quad (6)$$

$$y_{i+1} = b\, x_i$$

The map depends on two parameters a and b, which for the classical Henon map [8] have values of a = 1.4 and b = 0.3. For the classical values [8], the Henon map is chaotic.

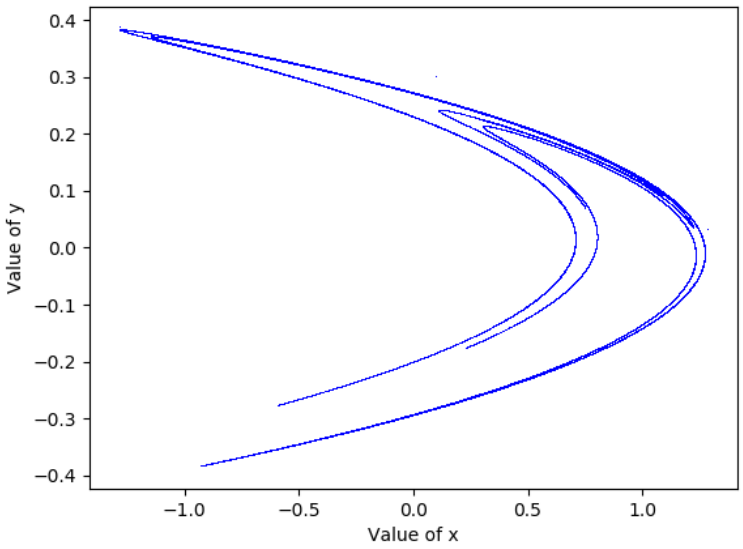


Figure 3.8 Exhibition of chaotic behavior of henon map with a=1.4 and b=0.3 along with initial values x=0.1 and y-0.3

3.5.1.1 Flow chart

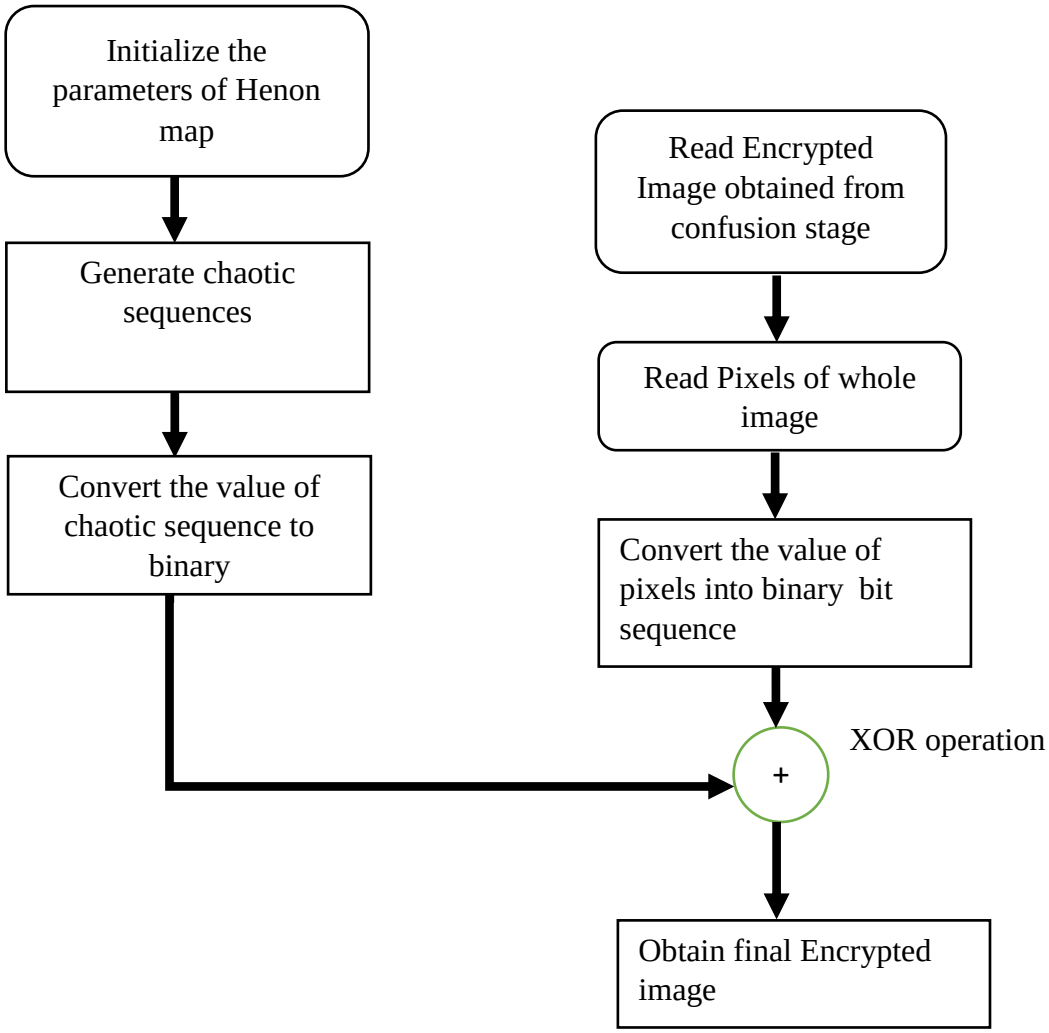


Figure 3.9 Flow chart of Image encryption using Henon map

3.6 Decryption

Here reverse algorithm of encryption has been used to get back the original image using the same chaotic maps with same initial conditions in different stages as the secret keys.

3.7 Security Analysis

The information entropy analysis, number of pixel change rate, Peak Signal to Noise Ratio and key sensitivity analysis has been be calculated for analyzing the performance of implemented algorithm .

3.7.1 Information Entropy Analysis

The construction of an image can be described using entropy that is considered a statistical degree of randomness. The entropy can be calculated as shown in equation [7].

$$E = -\Sigma_{i=0}^{N-1} P(X_i) \log_2 X_i \quad (7)$$

Where, P (X_i) represents the probability of occurrence of symbol.

3.7.2 Number of Pixel Change Rate (NPCR)

The ratio of the number of modifying pixels to the overall number of pixels is calculated by using NPCR that is known as the number of pixels modification amount of encrypted image while one pixel of original image is altered. NPCR can be calculated as shown in equation

$$NPCR = \frac{\Sigma_{I,J} D(i,j)}{WXH} X100 \quad (8)$$

Where, D(i,j) provides numbers of modifying pixels and WxH provides overall number of pixels in any image.

3.7.3 Peak Signal to Noise Ratio

Peak signal-to-noise ratio clarifies the divergence in pixel values amidst the original image and the image encrypted. An encryption planner is estimated by using PSNR that mirror the encryption goodness. The minimize value of PSNR acts preferable encryption goodness. PSNR can be calculated as in equation [9].

$$PSNR = 10\log_{10}\left[\frac{MXNX255^2}{\Sigma_{I=0}^{N-1}\Sigma_{j=0}^{M-1}[P(i,j)-C(i,j)]^2}\right] \quad (9)$$

Here, MXN is the total size of an image. P(i, j) is original image and C(i, j) is cipher image.

3.7.4 Key Sensitivity Analysis

Small alternation of key is very critical for a proper image encryption process. This means that a one-bit alternation in main key causes a very different result. Chaotic maps are extremely critical to control parameters and initial conditions. A decrypted image will not be identical to the original image if there is a tiny change in the major key [9].

3.7.5 Correlation Coefficient

The quantity of the disparity between two variables is calculated by correlation coefficient where it is performed to display the encryption goodness. The realization of the encryption process occurs when the Correlation Coefficient have to be very narrow to zero. Correlation coefficient is calculated by:

$$r_{x,y} = \frac{\mathrm{cov}(x,y)}{\sqrt{D(x)} \; X \sqrt{D(Y)}} \quad (10)$$

Where,

$$\mathrm{Cov}\,(x,y) = \frac{1}{N}\sum_{I=1}^{N}(x_i - E(x))(y_i - E(y)) \quad (11)$$

$$D(x) = \frac{1}{N}\sum_{I=1}^{N}(x_i - E(x))^2 \quad (12)$$

$$D(y) = \frac{1}{N}\sum_{I=1}^{N}(y_i - E(y))^2 \quad (13)$$

$$E(x) = \frac{1}{N}\sum_{I=1}^{N} y \quad (14)$$

$$E(y) = \frac{1}{N}\sum_{I=1}^{N} y \quad (15)$$

Where x and y are known as adjacent pixels of gray value, $r_{x,y}$ is correlation coefficient and COV is covariance of pixels. D(x) and D(y) are variances of x and y respectively and E(x) and E(y) are mean of x and y.

3.8 Assessment of visual cryptography

3.8.1 Structured Similarity Index Measurement (SSIM)

The difference with respect to other techniques such as MSE or PSNR is that these approaches estimate absolute errors; on the other hand, SSIM [11] is a perception-based model that considers image degradation as perceived change in structural information, while also incorporating important perceptual phenomena, including both luminance masking and contrast masking terms.

Let w_1 and w_2 be two windows from images I and I' respectively. The SSIM between the two windows w_1 and w_2 is shown in equation [16].

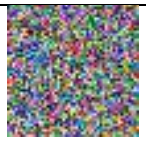
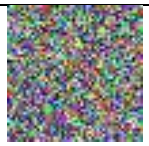
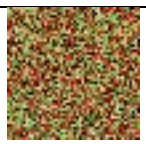
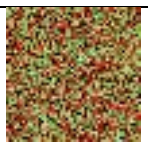
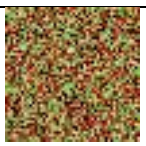
$$\mathrm{SSIM}(\mathrm{w}_1,\mathrm{w}_2)=\frac{(2\,\mu_{w1}\;\mu_{w2}+c_1)(2\sigma_{w1\,w2}+c_2)}{(\mu_{w1}^2+\mu_{w2}^2+c_1)(\sigma_{w1}^2+\sigma_{w2}^2+c_2)} \tag{16}$$

Where μ_{w1}, σ_{w1}, μ_{w2}, $\sigma_{w\,2}$ are the averages and variances of w_1 and w_2 respectively $\sigma_{w1\,w2}$ is the covariance of x and y and $C_1$and C_2 are constants.

Chapter Four: Result and Discussions

4.1 Results of implemented algorithm

The algorithm has been implemented using python programming language and the results have been observed on an Intel Core I5, 7th Generation computer with 8 GB RAM. The results of some experiments are given to prove efficiency and security of the implemented cryptosystem. The test images have been taken from image-net.

	Confusion stage			**Diffusion stage**
Original RGB Image [50x50]	Cubic map Secret keys r=3.67876 x=0.0145667685675567	Arnold cat map a=2,b=3 and n=8	Tent map r=1.62345 x=0.01256	Henon Map a = 1.4 b = 0.3 x0 = 0.1and y0 = 0.3
 tree.tif				
 aeroplane.png				
 peppers.tif				
 grassland.jpg				

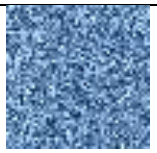
Original RGB Image [50x50]	Cubic map Secret keys r=3.67876 x=0.0145667685675567	Arnold cat map a=2,b=3 and n=8	Tent map r=1.62345 x=0.01256	Henon Map a = 1.4 b = 0.3 x0 = 0.1and y0 = 0.3
 flower.jpg				
 Volcano.jpg				

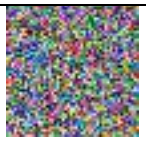
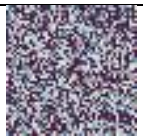
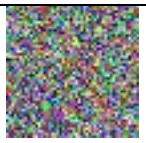
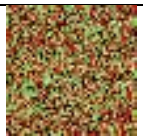
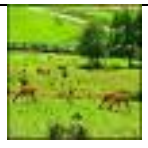
Figure 4.1 Results showing the image encryption in confusion and diffusion stage with the implementation of different chaotic maps

Here, six different test images in different formats (.jpg,.png and .tif)have been taken into account for the implementation of this algorithm. The image is divided into smaller blocks of 10x10.During the confusion stage, cubic map with secret keys r=3.67876 and x=0.0145667685675567has been implemented to obtain the encrypted image with shuffling of pixels with in the blocks. Similarly, Arnold cat map with a=2, b=3 and number of iteration=8 has been used to shuffle the blocks with in the whole image. Tent map with secret key r=1.62345 and x=0.01256 performs the shuffling of pixels with in the image. Ultimately, during the diffusion stage, Henon map with initial parameters a = 1.4 b = 0.3 x_0 = 0.1 and y_0 = 0.3 has been implemented to change the value of pixels with the help of chaotic sequences generated with the given parameters.

4.2 Results of Performance analysis

4.2.1 Key sensitivity analysis

An ideal image encryption algorithm should be sensitive with respect to both the input secret key and original image. The change of a single bit in either the secret key or should produce completely different output results. To prove the robustness of the implemented algorithm, sensitivity analysis has been performed with respect to key and image. High key sensitivity is required by secure image cryptosystem, which means that the encrypted image cannot be decrypted correctly although there is only a slight difference between secret key. This guarantees the security of the implemented algorithm against brute-force attacks to some extent. For testing the key sensitivity of the implemented algorithm, following steps have been performed:

Encrytped Image	Decryption with change in initial key i.e r=3.67875 and x=0.0145667685675566	Decryption with same initial key i.e. r=3.67876 and x=0. 0145667685675567
		 tree.tif
		 aeroplane.png
		 peppers.tif
		 grassland.jpg

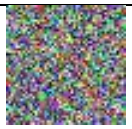
Encrytped Image	Decryption with change in initial key i.e r=3.67875 and x=0.0145667685675566	Decryption with same initial key i.e. r=3.67876 and x=0. 0145667685675567
		 flower.jpg
		 Volcano.jpg

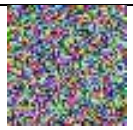
Figure 4.2 Key sensitivity analysis with slight change in the secret key

4.2.2 Information Entropy Analysis

The information entropy values of final encrypted images after the diffusion stage have been obtained to signify the notion of randomness of pixels.

Table 1 Information Entropy Analysis for Red, Green and Blue pixels in the final encrypted image.

Images	**Entropy**
 tree_henon_encrypted.png	7.97 [R] 7.98 [G] 7.94 [B]
 aeroplane_henon_encrypted.png	7.93 [R] 7.95 [G] 7.97 [B]
 peppers_henon_encrypted.tif	7.97 [R] 7.96 [G] 7.92 [B]

Images	Entropy
 grassland_henon_encrypted.jpg	7.93 [R] 7.99 [G] 7.90 [B]
 Flower_henon_encrypted.jpg	7.94 [R] 7.96 [G] 7.97 [B]
 Volcano_henon_encrypted.jpg	7.95[R] 7.98[G] 7.96[B]

The information entropy analysis for RGB pixels in final encrypted image gives greater results of information entropy. The values obtained by the experiments are very close to the theoretical value of 8**.** So, it is pertinent to say that the information leakage in the image enciphering process is negligible and the implemented cryptosystem is robust upon the entropy attack.

4.2.3 Number of Pixel Change Rate (NPCR):

Number of pixel changed rate (NPCR) compares the pixel values of the original image and the encrypted image. The resultant value has been returned in percentage.

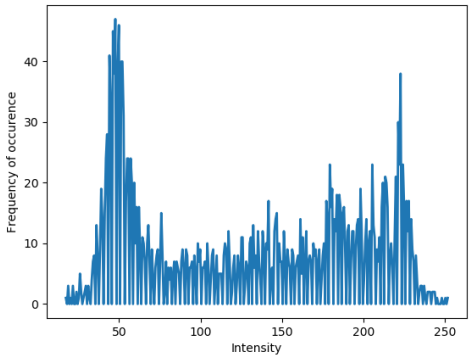
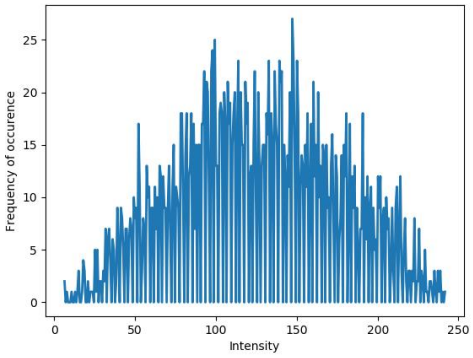
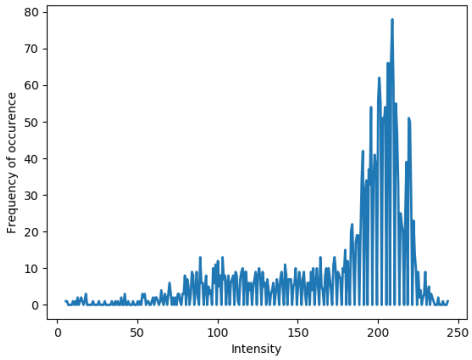
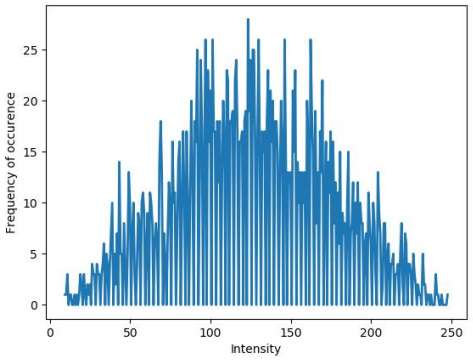
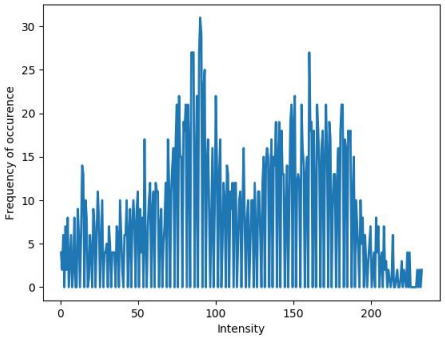
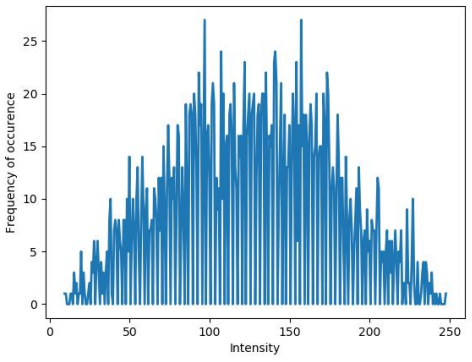
Table 2 Number of Pixel change Ratio analysis of final diffused image obtained after the implementation of multiple chaotic maps.

Original Images	**Number of Pixel Change Ratio (%)**
Tree.tif	99.73
Aeroplane.png	99.64
Peppers.tif	99.62
Grassland.jpg	99.46
Flower.jpg	99.58
Volcano.jpg	99.60

The analysis is positive since the number of pixel change at the end of diffusion stage is more than 99% .The result shows the increment in the NPCR from confusion stage to diffusion stage which signify the use of chaotic maps.

4.2.4 .Histogram Analysis

The information of image at the confusion stages doesn't changes. So, histogram resembles with the histogram of original image. But during transmission of cipher image after diffusion stage, it doesn't contain exact information of original image. It has been depicted in given histogram that there is no any similarity in original and final diffused image.

Original Image [50x50]	**Diffusion stage**
Tree.tif 	
Aeroplane.png 	
Peppers.tif 	

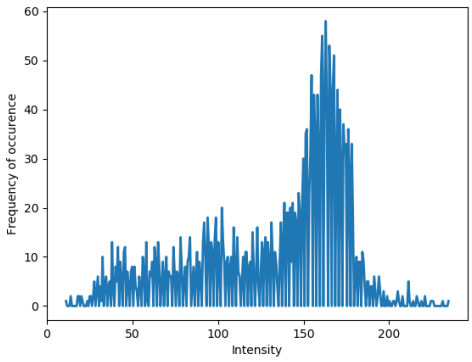
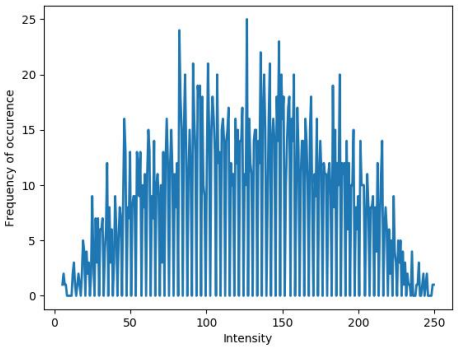
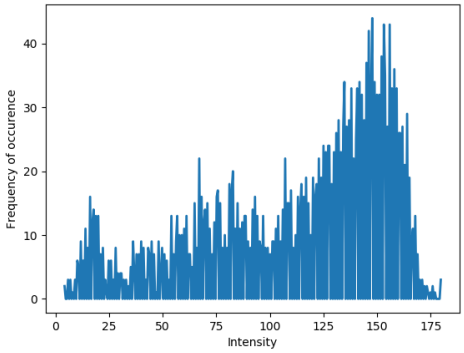
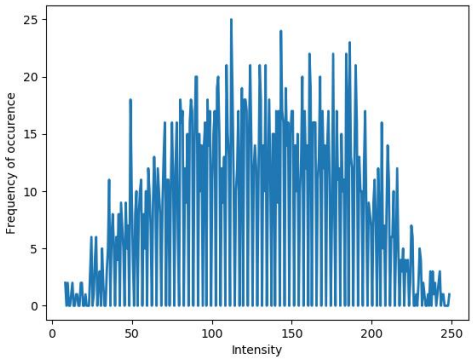
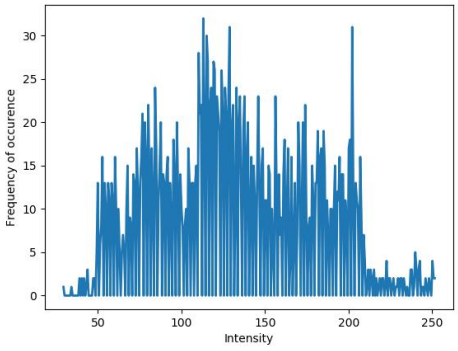
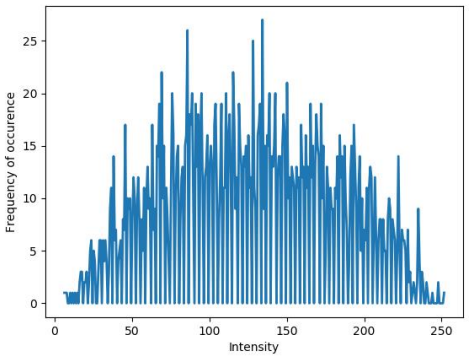
Original Images	**Diffusion stage**
Grassland.jpg 	
Flower.jpg 	
Volcano.jpg 	

Figure 4.3 Histogram analysis of original image and encrypted image at diffusion stage.

4.2.5 Key space analysis

This image cryptosystem employs different keys for different chaotic maps. Cubic map uses two keys (i.e. x_0 and r),Arnold cat map uses three keys(i.e. a, b and N),Tent map uses (x_0 and r) as secret keys during confusion stage .Similarly, Henon map employs two keys (x_0,y_0) during diffusion stage for image encryption.. The implemented algorithm uses 2^{128} different combinations of the secret keys for cubic map during implementation**.** An image encryption with such a long key space is sufficient for reliable practical use.

4.2.6 Correlation Analysis

Each pixel in an ordinary image is highly correlated with its adjacent pixels either in horizontal or vertical or diagonally. An ideal encryption design should produce cipher images with no such correlation to the adjacent pixels. Here correlation coefficients for horizontally, vertically and diagonally adjacent pixels have been computed respectively. Table 3 shows the correlation of the plain image whereas Table 4 shows the correlation between final encrypted cipher images. It can be easily found that the correlation of the initial image is an obvious linear relationship, whereas the correlation of the cipher image shows a stochastic relationship.. The results show that the correlation coefficients of the plain image are all close to 1.Similarly, the correlation coefficients of the cipher image is close to 0. This indicates that the implemented encryption algorithm possesses high security against statistical attacks.

Table 3 Correlation coefficients of neighborhood pixels at different directions of original images

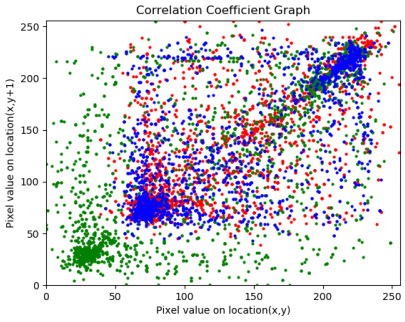
Original Image	**Horizontal**	**Vertical**	**Diagonal**
Tree.tif	0.694	0.835	0.647
Aeroplane.png	0.853	0.985	0.801
Peppers.tif	0.892	0.856	0.857
Grassland.jpg	0.986	0.826	0.776
Flower.jpg	0.876	0.931	0.854
Volcano.jpg	0.873	0.903	0.857

Table 4 Correlation coefficients of neighborhood pixels at different directions of final encrypted cipher images.

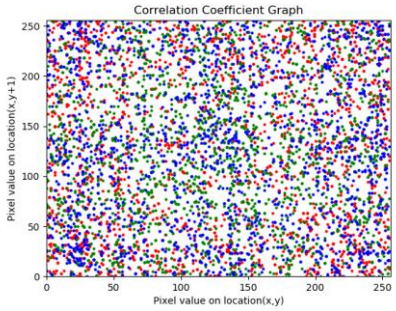
Final Encrypted Image	**Horizontal**	**vertical**	**Diagonal**
Tree_henon.tif	-0.0144	0.0242	-0.0142
Aeroplane_henon.png	-0.0255	-0.0427	-0.0006
Peppers_henon.tif	-0.0062	-0.0277	-0.0113
Grassland._henon.jpg	0.0316	0.0826	0.0231
Flower_henon.jpg	0.0583	0.0145	0.0181
Volcano_henon.jpg	0.0772	0.0563	0.0721

:

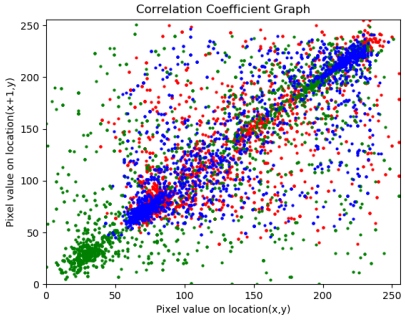
4.2.6.1 Correlation Plot



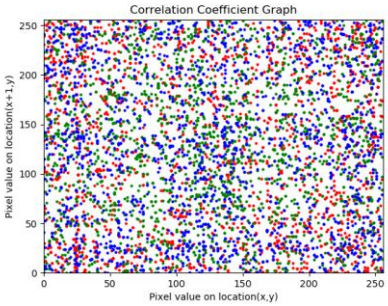
(a)



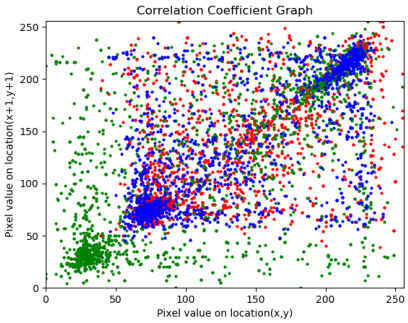
(b)



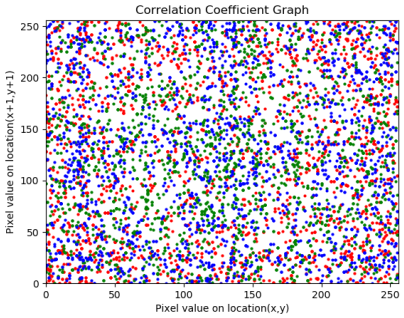
(c)



(d)

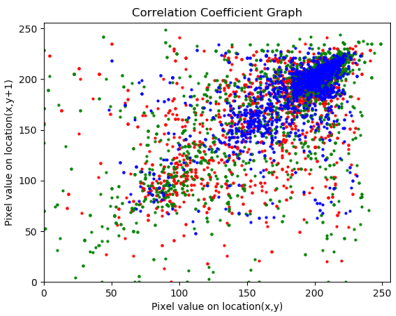


(e)

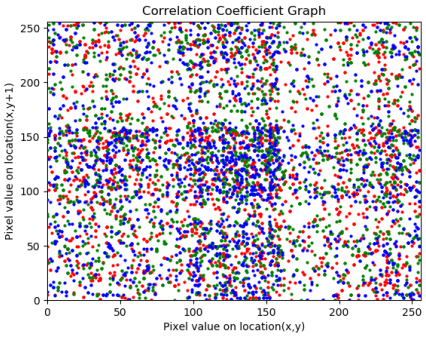


(f)

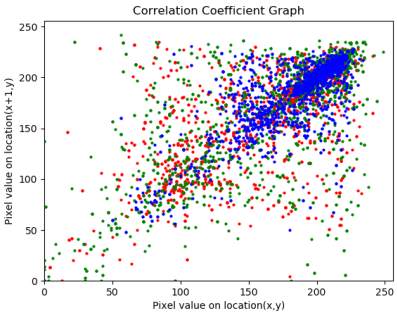
Figure 4.4 Correlation plot of horizontally ,vertically and diagonally associated pixels of original image (a),(c) and (e) and correlation plot of horizontally ,vertically and diagonally associated pixels of final encrypted image (b),(d) and (f) of Tree.tif



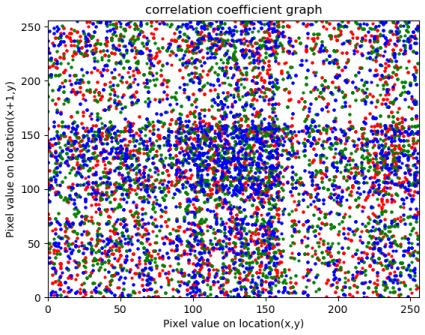
(a)



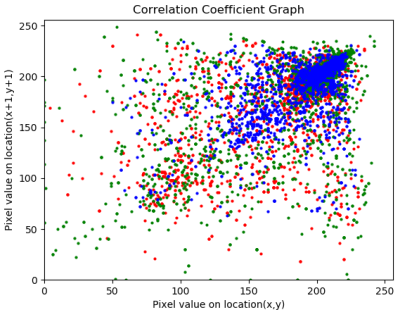
(b)



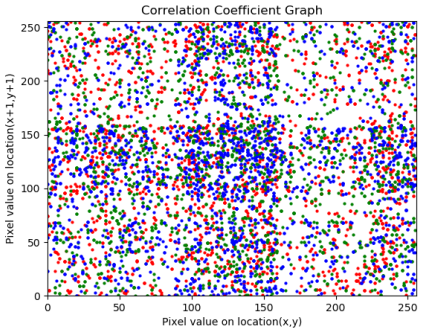
(c)



(d)

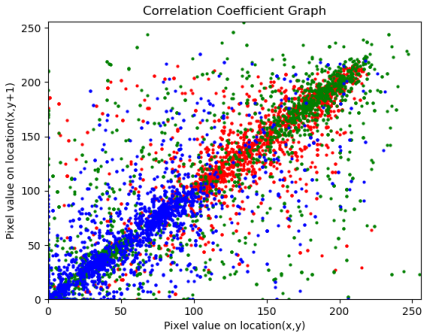


(e)

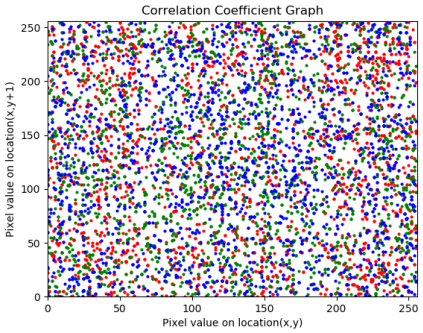


(f)

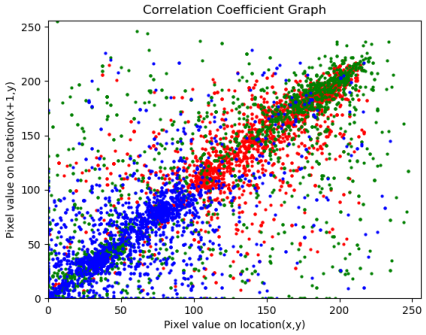
Figure 4.5 Correlation plot of horizontally ,vertically and diagonally associated pixels of original image (a),(c) and (e) and correlation plot of horizontally ,vertically and diagonally associated pixels of final encrypted image (b),(d) and (f) of aeroplane.jpg



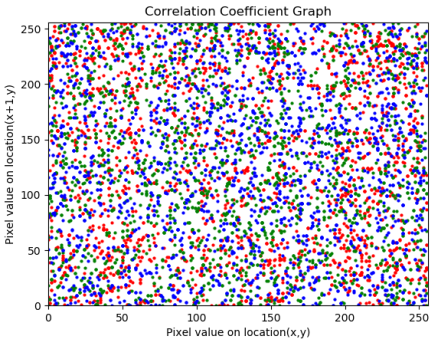
(a)



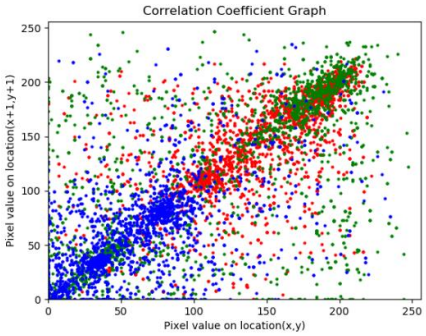
(b)



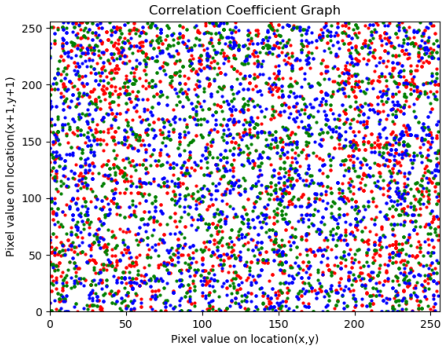
(c)



(d)

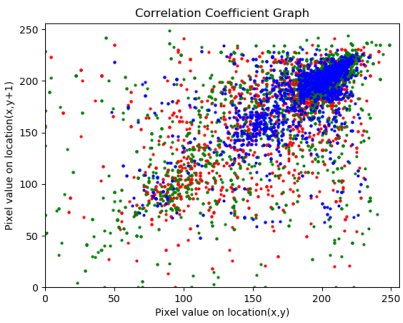


(e)

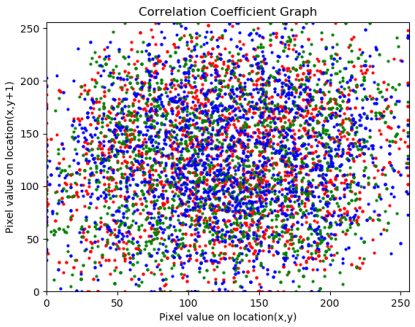


(f)

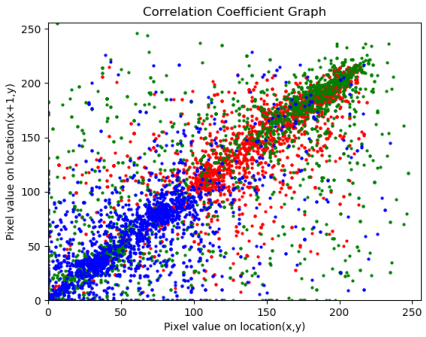
Figure 4.6 Correlation plot of horizontally ,vertically and diagonally associated pixels of original image (a),(c) and (e) and correlation plot of horizontally ,vertically and diagonally associated pixels of final encrypted image (b),(d) and (f) of Peppers.tif.



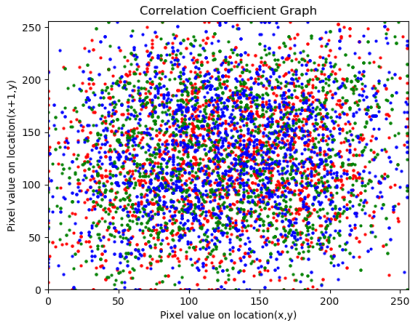
(a)



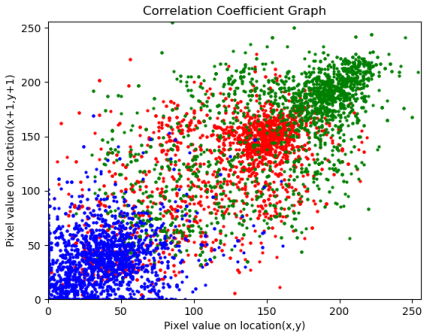
(b)



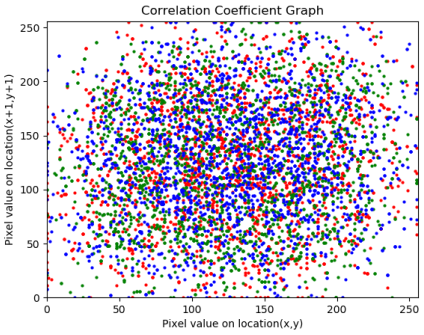
(c)



(d)

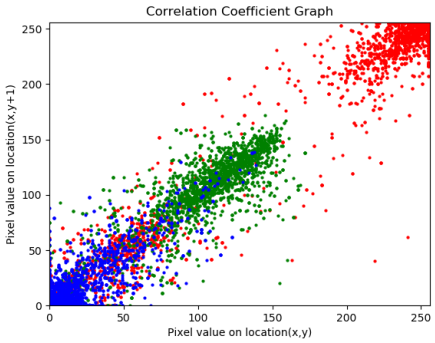


(e)

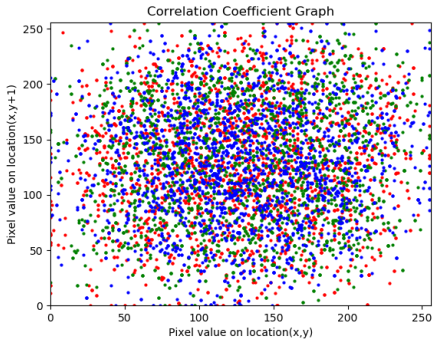


(f)

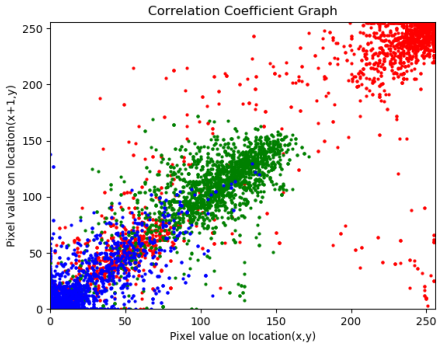
Figure 4.7 Correlation plot of horizontally ,vertically and diagonally associated pixels of original image (a),(c) and (e) and correlation plot of horizontally ,vertically and diagonally associated pixels of final encrypted image (b),(d) and (f) of grassland.jpg



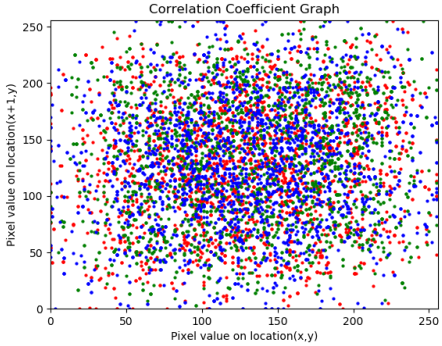
(a)



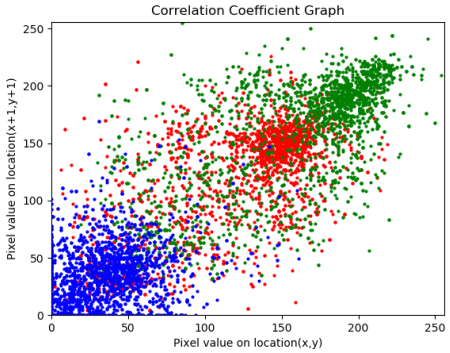
(b)



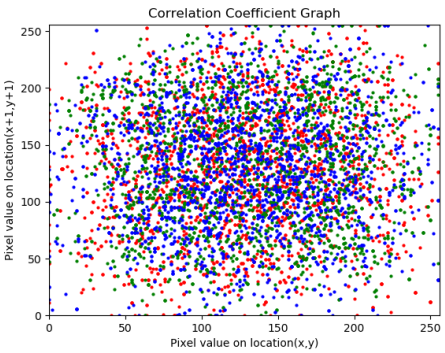
(c)



(d)

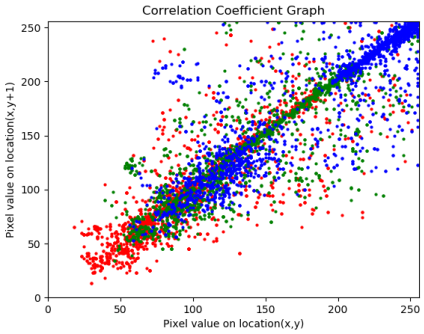


(e)

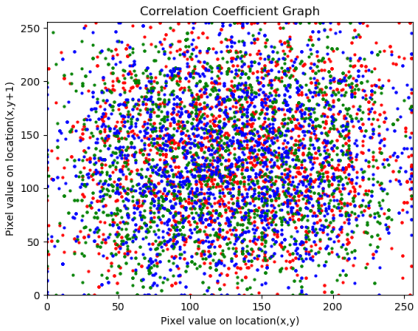


(f)

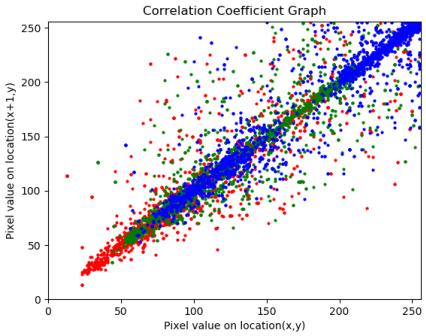
Figure 4.8 Correlation plot of horizontally ,vertically and diagonally associated pixels of original image (a),(c) and (e) and correlation plot of horizontally ,vertically and diagonally associated pixels of final encrypted image (b),(d) and (f) of of flower.jpg



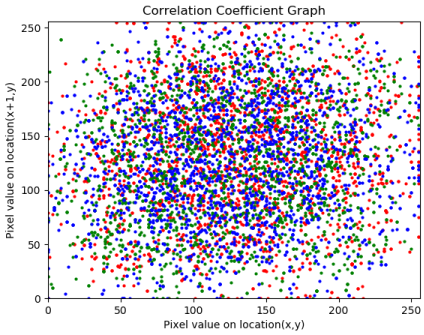
(a)



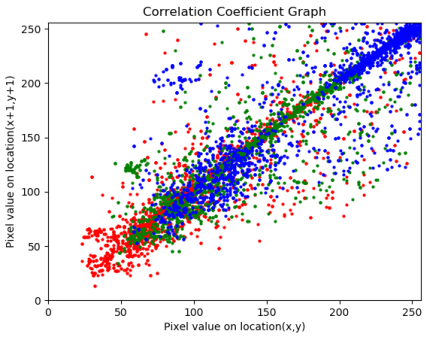
(b)



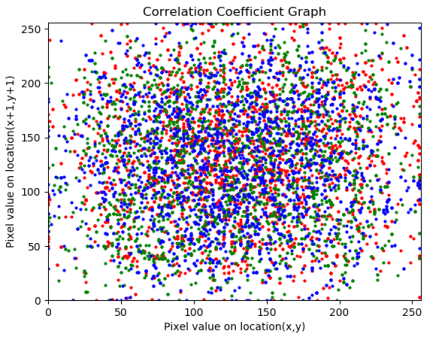
(c)



(d)



(e)



(f)

Figure 4.9 Correlation plot of horizontally ,vertically and diagonally associated pixels of original image (a),(c) and (e) and correlation plot of horizontally ,vertically and diagonally associated pixels of final encrypted image (b),(d) and (f) of Volcano.jpg

.

4.2.7. Structural Similarity Index Measurement (SSIM)

Structural Similarity Index measurement (SSIM) has been performed to evaluate visual quality between original image and final encrypted image. The index depicts that there is complete dissimilarity between original image and final encrypted image. Similarly the measurement Peak Signal to Noise Ratio (PSNR) of original image and final encrypted image exhibits the disassociation of pixels in the image after diffusion stage using Henon map.

Table 5 Exhibit SSIM and PSNR between original image and final encrypted image

Images	SSIM between original image and final encrypted image	PSNR between original image and final encrypted image (dB)
Tree.tif	0.0035	26.82
Aeroplane.png	0.0152	27.84
Peppers.tif	0.0143	27.94
Grassland.jpg	0.0051	27.92
Flower.jpg	0.0026	27.84
Volcano.jpg	0.0046	27.89

Chapter Five: Conclusion

5.1 Conclusion

In this thesis, efficient chaos based image cryptography has been implemented to address the security flaws of the traditional image cryptosystems. This algorithm combines good confusion and diffusion properties by using chaotically coupled chaotic maps. It utilizes chaotic confusion of image pixels by using chaotic coupling between chaotic maps and changing the pixel values in diffusion process at the time of image transmission. The average entropy value of 7.97 provides vigorous upon entropy attack. The Number of Pixel Change Ratio (NPCR) of above 99% obtained from the diffusion stage signify the use of multiple chaotic maps. The image cryptosystem has 2^{128} different combinations of the secret keys. The maximum correlation values in vertical, horizontal and diagonal directions of encrypted images are 0.0826, 0.0772 and 0.0721 respectively and structured similarity index measurement of not more than 0.0152 between original image and encrypted image validates the robustness of this algorithm.

Therefore, these results lead to the effectiveness and robustness of the implemented image cryptosystem. The encryption and decryption algorithms are symmetric, making it suitable for multimedia encryption. Due to the excellent confusion and diffusion properties, both exploiting important properties of chaos, the cryptosystem is extremely secure and fast enough to be used in real-time communications.

5.2 Future Enhancements

This image cryptosystem can be further enhanced by reducing the size of blocks of pixels from 10x10 to smaller block size. Furthermore, a novel chaos based image cryptosystem can be also be realized by mapping multiple chaotic maps beside than the mentioned ones in this research work.

References

[1] Chenghang Yu, Baojun Zhang and Xiang Ruan(2011),The Chaotic Feature of Trigonometric Function and Its Use for Image Encryption, Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD).

[2] C.K. Huang and H.H. Nien(2009), Multi chaotic systems based pixel shuffle for image encryption, Optics Communications 282 (2009) 2123–2127.

[3] Shubo Liu, Jing Sun1, Zhengquan Xu(2009), An Improved Image Encryption Algorithm based on Chaotic System, Journal of Computers, Vol. 4, No. 11.

[4] Hamid Nejati ,Ahmad Beirami, Yehia Massoud ,A realizable modified tent map for true random number generation, 1558-3899, 10-13 Aug. 2008

[5] LEI Li-hong ,BAI Feng-ming,HAN Xue-hui(2013), New Image Encryption Algorithm Based on Logistic Map and Hyper-chaos, International Conference on Computational and Information Sciences.

[6] John Justin M, Manimurugan S (2012), A Survey on Various Encryption Techniques, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012.

[7] Komal D Patel, Sonal Belani(2011),Image Encryption Using Different Techniques Techniques:A Review, International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, Volume 1, Issue 1, November 2011).

[8] Rashidah Kadir, Rosdiana Shahril, Mohd Aizaini Maarof (2010), 6 A modified image encryption scheme based on 2D chaotic map, International Conference on Computational and Information Sciences.

[9] XiaoJun Tong , Yang Liu, Miao Zhang and Zhu Wang , A Novel Image Encryption Scheme Based on Dynamical Multiple Chaos and Baker Map, IAENG International Journal of Computer Science, 35:1, IJCS_35_1_03.

[10] Hooman Kashanian Masoud Davoudi and Hamed Khorramfar, Image Encryption using chaos functions and fractal key, IJCSNS International Journal of Computer Science and Network Security, VOL.16 No.10, October 2016

[11] Maged Wafy and Rasha Ebaid ,Quality Assessment of Visual Cryptography Images, Australian journal of basic and applied sciences ISSN:1991-8178 ,October 2016.

Appendix

1. Code to encrypt image using cubic map

```
im=Image.open('Image.jpg')
pix=im.load()
l,h =im.size
pixels  = {}
for x in range(l):
      for y in range(h):
         pixel=pix[x,y]
         n=l*x+y
         pixels.update({n:pixel})
# use cubic map to generate random number
x=0.01456
c_numbers=[]
for a in range (l*h):
        x1= 3.67876 * x**3 + (1 - 3.67876)*x
        x=x1
        c_numbers.append((a,x1))
# short random numbers along with key
p=sorted(c_numbers,key=lambda  l:l[1])
#split key list
q=[i[0] for i in p]
a=0
imgg =Image.new('RGB', (l, h), "black")
points=imgg.load()
for i in range(imgg.size[0]):
    for j in range (imgg.size[1]):
        points[i,j]= pixels [q[a]]
```

```
        a+=1
imgg.save('Image_cubic.jpg')
```

2. Code to encrypt the image obtained from cubic map using Arnold cat map

```
im = Image.open('image_cubic.jpg')
width, height = im.size
unit = width/10
blocks = {}
r=0
#print width
for j in range(10):
    for i in range(10):
        area=(unit*i,unit*j, unit*(i+1),unit*(j+1))
        r=j*10+i
        img = im.crop(area)
        blocks.update({r:img})
sn=[]
for k in blocks.keys():
  sn.append(k)
print (sn)
ns = []
#arnold cat map for block shuffling
def arnold_cat(n):
    ns = []
    for num in n:
        p= num%10
        q= int(num/10)
        q = (q+p) % 10
        p = (2*q+3*p) % 10
        ns.append(q*10+p)
    return ns
key=5
```

```
n = sn
for i in range(key):
    m=arnold_cat(n)
    n=m
encr=[]
for p in n:
    encr.append(blocks[p])
imgg =Image.new('RGB', (width, height), "white")
temp=imgg
i=0
for a in range(10):
    for b in range(10):
        loc=[22*b,22*a]
        temp.paste(encr[i],loc,None)
        i+=1
temp.save("image_arnold.jpg")
```

3. Code to encrypt the image obtained from Arnold cat map using Tent map

```
im=Image.open('image_arnold.jpg')
pix=im.load()
l,h =im.size
pixels  = {}
for x in range(l):
     for y in range(h):
         pixel=pix[x,y]
         n=l*x+y
         pixels.update({n:pixel})
# use tent map to generate random number
c_numbers=[]
for a in range (lxh):
       if x < 0.5:
          xl = 1.67584 * x
          x=xl
```

```
        elif x > 0.5:
            xl = 1.67584 - 1.67584 * x
            x=xl
            print(x)
c_numbers.append((a, xl))
# short random numbers along with key
p=sorted(c_numbers,key=lambda  l:l[1])
print()
#split key list
q=[i[0] for i in p]
a=0
imgg =Image.new('RGB', (l, h), "black")
points=imgg.load()
for i in range(imgg.size[0]):
    for j in range (imgg.size[1]):
        points[i,j]= pixels [q[a]]
        a+=1
imgg.save('image_tent.jpg')
```

4. Code to perform final encryption using Henon map in diffusion stage

```
read_image =Image.open('image_tent.jpg','r')
plain_image=list(read_image.getdata())

pix=im.load()
l,h =im.size
a = 1.4
b = 0.3
x0 = 0.1
y0 = 0.3
def henon_map(x,y):
    x1= y + 1.0 - a *x*x
    y1= b * x
    return x1,y1
```

```
cipher_image=[]
sum=0
for p in range(lxh):
    tup=()
    for r in range(3):
        rgb = plain_image[p][r]
        block =0
        for s in range(8):
            x1, y1 = henon_map(x0,y0)
            if x1 < 0.3992:
                bit =0
            else:
                bit =1
            block+=bit*(2**s)
            x0=x1
            y0=y1
        cipher_rgb =rgb^block
        tup+=(cipher_rgb,)
    cipher_image.append(tup)
im2 = Image.new(read_image.mode, read_image.size)
im2.putdata(cipher_image)
im2.save('Final_encrypted_Image.jpg')
```