Hybrid Image Encryption Technique Using Genetic Algorithm and Lorenz Chaotic System

Vishwanath Chikkareddi[1,*], *Anurag* Ghosh[2,**], *Preksha* Jagtap[3,***], *Sahil* Joshi[4,****], and *Jeel* Kanzaria[5,†]

[1]Ramrao Adik Institute of Technology
[2]Navi Mumbai
[3]Maharashta

Abstract. One of the important application of image encryption is storing confidential and important images on a local device or a database in such a way that only the authorized party can view or perceive it. The current image encryption technique employs the genetic algorithm to increase confusion in the image, but compromises in time and space complexity. The other method employs chaos or pseudo random number generating systems which have fast and highly sensitive keys but fails to make the image sufficiently noisy and is risky due to its deterministic nature. We propose a technique which employs the non-deterministic, optimizing power of genetic algorithm and the space efficiency and key sensitivity of chaotic systems into a unified, efficient algorithm which will retain the merits of both the methods whereas tries to minimize their demerits in a software system. The encryption process proceeds in two steps, generating two keys. First, an encryption sequence is generated using Lorenz Chaotic system of differential equation. The seed values used are the user's actual key having key sensitivity of 10^{-14}. Second, the encrypted image's genetic encryption sequence is generated which will result in an encrypted image with entropy value greater than 7.999 thus ensuring the image is very noisy. Proposed technique uses variations of Lorenz system seed sets to generate all random mutations and candidate solutions in Genetic encryption. Since only the seed sets leading to desired solution is stored, space efficiency is higher compared to storing the entire sequences. Using this image encryption technique we will ensure that the images are hidden securely under two layers of security, one chaotic and other non-deterministic.

1 Introduction

Nowadays, the concern of information security is gaining immense importance, hence the security of confidential information and data in the field of Computer Science is crucial. In the 21th century, images have become an integral part of digital information hence maintaining the integrity, access control and confidentiality of image data from unauthorized users are essential. Cryptography is a vital tool for guarding such information in computer systems. It refers to secure information and communication techniques that descended from mathematical concepts along with a set of algorithms to reconstruct messages in ways that are hard to decipher. Encryption is translation of data into another form, or code so that only people with access to a decryption key or password can access it. Image encryption plays a crucial role in the field of information security and makes the information in the image unreadable. Consequently, no third party can decrypt and retrieve information from this image, thus ensuring complete privacy for the users. In the past years, a lot of research has been made into refining algorithms that deal with data encryption. The genetic algorithm is one such type of non-deterministic evolutionary algorithm which uses its powerful features like crossover and mutation to generate a symmetric key which also augments the entropy furthermore the noise of the resulting encrypted image [4]. Chaos-based cryptographic systems have become an important part of data encryption techniques because of their excellent performance in generating highly sensitive keys and prompt results [1]. Our aim is to propose an integrated algorithm which employs the agility, efficiency and high key sensitivity of chaotic systems and the optimizing power of the genetic algorithm. In this paper, a novel method of image encryption is introduced and the aim to design a system for optimizing and analyzing the whole process of image encryption wherein Lorenz Chaotic System and Genetic Algorithm are employed to encrypt and store the image data securely is achieved.

2 Literature Survey

Encryption is a process of encoding the data to prevent unauthorized access. With the rapid growth of computer networks, the world we live in sharing images has become a major part of our lives. Privacy is of utmost importance. Hence we need image encryption to secure the image for

*e-mail: vishwa.chikareddy@gmail.com
**e-mail: ghoshanurag99@gmail.com
***e-mail: prekshajagtap@gmail.com
****e-mail: joshi.k.sahil@gmail.com
†e-mail: jeelkanzaria1997@gmail.com

unauthorized use. Different papers to propose an efficient method for image encryption have been referred.

The paper[1] proposed a inventive chaotic picture encryption method dependent on the opposite fractal insertion function. This framework is connected to produce clamorous arrangements. The groupings inferred are then used to permute the pixel positions to get the rearranged picture by disorganized succession arranging. To upgrade the security the acquired chaotic sequences are then evaluated and used to perform dissemination. A new scheme based on mixed chaotic map and Josephus traversing is being proposed in methodology [2]. The proposed technique comprises three procedures. Initially a key stream generator is structured dependent on another proposed plan of chaotic frameworks. At that point Josephus traversing is utilized in scrambling, next, the segments and the columns of the pixels are traded in extent to characterized guidelines lastly the chaotic coordinates are utilized to exchange the places of every pixel. In the last advance the picture information and four disordered maps are utilized to change the pixel gray level values and break the solid relationships between neighboring pixels of the picture at the same time. The consequences of this technique and security investigation affirm that the framework proposed has better execution. A picture cryptography framework dependent on a chaotic True Random Bit Generator (TRBG) is anticipated in method [3]. The chaotic generator might be a nonlinear electronic circuit that produces twofold parchment chaotic attractors. The estimations of the underlying conditions rationale entryway parameters are the keys of the cryptological framework. Since the dynamic conduct of the circuit is incredibly unusual, a genuine arbitrary bits succession is made through perceived procedure. Likewise, the assurance investigation of the scrambled picture represents the high security of the anticipated topic. A hybridized model for encoding pictures through a blend of hereditary calculation and DNA Sequence was anticipated in procedure [4]. The encoding method comprises two stages Transportation and Scrambling area and Substitution stage. In the initial stage, the pixel areas are modified by utilizing GA to curtail the relationship among nearby pixels. In the substitution part, pixels are supplanted by utilizing XOR activity between the pixel esteems redesigned into double strings and DNA corrosive sub-strings got from an irregular DNA string. The DNA sub-strings are utilized as keys for picture encryption. The exploratory result approves that the calculation is direct, quick and conceivable. A usage of computerized picture encoding framework utilizing the Lorenz chaotic framework is anticipated in strategy [5]. Subsequent to creating a disorganized key stream, the hash estimation of plain picture is inserted to change the initial secret keys progressively to help security. The anticipated digital picture cryptography calculation basically contains two stages. In the primary stage we will in general create secret keys utilizing 256-piece hash estimation of plain picture and furthermore the strategic guide. Inside the second segment the plain picture is encoded by utilizing Lorenz confused arrangements with the key keys from the past segment. The projected method [6] uses adjustive Genetic algorithm to create an efficient variety of optimum clusters centers in real time pictures. Genetic operators like mutation and crossover are done and are enforced with an adjustive nature to try and do work expeditiously to decide the cluster heads. By using this data K-means clustering rule is applied. The output is a clustered medical look. Through the results it may be concluded that the projected technique of clustering of medical pictures using genetic rule is effective. Segmentation is needed to dis-join a complete image into tiny segments such that they collectively cover the complete image. This was planned in approach [7]. within the planned system, we tend to begin by taking a gray image. Noise is introduced during this image using numerous noise models. Afterwards Genetic algorithm is applied for segmentation of the obtained image. Thus, the final segmented image is obtained and at last we tend to confirm the peak Signal to Noise ratio and Mean square Error for noisy image and metameric image. The values obtained are compared with alternative image segmentation algorithmic programs like OSTU and Watershed and it's concluded that Genetic algorithm works best for segmentation of noisy image. An investigation on the employment of other kinds of sequences referred to as chaotic sequences for DS-SS system is planned in procedure [8]. It also thinks about the execution of picture encoding utilizing run of the mill pseudo irregular code generators thereto utilizing new confused sequence generator. A buildup arithmetic number is added to the framework; this strategy is assessed and thought about thereto of non residue numeration framework and it quantifies its execution. To add a great deal of highlights to the riotous correspondence framework Residue numeration framework (RNF) is included. The creators of [9] propose a standard for pseudo irregular range arrangement generator utilizing tests of Chen chaotic framework. This procedure explains the matter of the non-uniform possibility dispersion of grouping created specifically by the Chen disordered framework. The varied applied arithmetic tests check that the arrangements created by our anticipated framework have reasonable factual properties. The test consequences of the security examination approve the high ability of the anticipated system to oppose changed assaults. Further depictions of the use of pseudo random number generators in the domain of image encryption techniques with the incorporation of a perceptron model within a neural network proposed in [10] and the used of transposition and substitution techniques in [11] is used to further the encryption efforts to ensure the abstraction of the information to a great extent. The proposed methodology in the work in [13] features the uses of multiple pseudo random generator techniques in various steps of the mentioned encryption technique which are the chaotic map functions. The used functions being Cubic Map, Henon Map, Quadratic Map, Logistic Map. The work has as mentioned in the paper proved to have better encryption quality in terms of entropy and correlation than the compared works.

3 Proposed Methodology

In the proposed methodology, image encryption algorithm takes part in two major phases using two different techniques of random sequence generation and genetic algorithm (G.A.) to optimize/improve image noise. They are encryption using Lorenz chaotic system and noise enhancement using genetic algorithm. In the first phase we generate a random sequence of integers $K_c = \{k_{c_0}, k_{c_1}, k_{c_2}, k_{c_3}, ..., k_{c_n}\}$ whose values range form [0...255] corresponding to the first layer of encryption. This sequence K_c is generated using Lorenz Chaotic system of Differential equations. The length of sequence K_c i.e. n will be equal to the number of pixels in the image. Each pixel in the original image I is traversed and sequentially XORed with the values in the sequence K_c generating an intermediate encrypted image I_{en_1}. The next phase takes the image I_{en_1} as input to the Genetic algorithm and generates a genetic encryption sequence $K_g = \{k_{g_0}, k_{g_1}, k_{g_2}, k_{g_3}, ..., k_{g_n}\}$ which is another sequence of integral values where individual value ranges from [0...255] having length equal to the number of pixels in the image. These values are again sequentially XORed onto each pixel of the image I_{en_1} and the final encrypted image I_{en_2} is generated. For decryption the sequences K_c and K_g are regenerated, whose individual values are sequentially XORed to each image pixel of image I_{en_2} to decrypt and regenerate the original image I.

3.1 Encryption Process With Lorenz Chaotic System:

The Lorenz system, first studied by Edward Lorenz around 1960, is a dynamical system described by the following nonlinear system of ordinary differential equations:

$$\frac{dx}{dt} = a(y - x), \tag{1}$$

$$\frac{dy}{dt} = (c - z)x - y, \tag{2}$$

$$\frac{dz}{dt} = xy - bz \tag{3}$$

The real numbers given by a, b, c are the control parameters, whereas real values given by x, y, z are called the state variables, and the given equations 1, 2 and 3 are for the time derivatives of the variables x, y and z. For a given set of control parameters and initial set of values x_0, y_0 and z_0 which are the initial state variables are provided. All these values are seed values and combined form a sets call as the seed set which is the encryption, decryption key for this phase. The system is non linear , non periodic. It does not repeat its values over a time period and takes three input variables x y and z. This implies that the system is three dimensional and deterministic thus from a given current state and inputs, it can always be determined what the next state will be.

Step 1: The generation of the random sequence is done by first defining a key /seed set containing the seed values for the Lorenz system as follows:

$$x_0 = 15 + random(-1, 1) * random(10^{-14}, 10^{-2}) \tag{4}$$

$$y_0 = 28 + random(-1, 1) * random(10^{-14}, 10^{-2}) \tag{5}$$

$$z_0 = 8/3 + random(-1, 1) * random(10^{-14}, 10^{-2}) \tag{6}$$

Here the random() function takes two real numbers as parameters random(m,n) where m is the lower limit and n is the higher limit, random() generates a real number v such that, m ≤ v ≤ n. Similarly we consider control parameters as

$$a = 22 + random(-1, 1) * random(10^{-14}, 10^{-2}) \tag{7}$$

$$b = 9.0 + random(-1, 1) * random(10^{-14}, 10^{-2}) \tag{8}$$

$$c = 12 + random(-1, 1) * random(10^{-14}, 10^{-2}) \tag{9}$$

Also, initially we have to define the number of steps after which we will extract values from the Lorenz system

$$dt = 0.01 + random(-1, 1) * random(10^{-14}, 10^{-4}) \tag{10}$$

Step 2: These values a, b, c, x_0, y_0, z_0 and dt are the seed values and the set of values that users needs to store for generating encryption sequence K_c instead of storing the entirety of K_c. Once these parameters are set i.e. a user key is defined, we begin our iteration to generate sequence K_c. Consider an image I having height H and width W that means H pixels in a row and W pixels in a column respectively, the number of random numbers to be generated are H * W. That means H * W values or samples of x , y and z are considered.

Let x_{i-1}, y_{i-1} and z_{i-1} be the values of x, y and z at iteration i-1 respectively. The values for x, y and z in the next iteration is given by:

$$x_i = x_{i-1} + dx_i, \tag{11}$$

$$y_i = y_{i-1} + dy_i, \tag{12}$$

$$z_i = z_{i-1} + dz_i, \tag{13}$$

Substituting the values of dx, dy and dz from equations 1, 2 and 3 we get

$$x_i = x_{i-1} + a(y_{i-1} - x_{i-1})dt, \tag{14}$$

$$y_i = y_{i-1} + ((c - z_{i-1})x_{i-1} - y_{i-1})dt, \tag{15}$$

$$z_i = z_{i-1} + (x_{i-1} - b * z_{i-1})dt \tag{16}$$

To generate random whole number less than K. Let S be a key set such that

$$S = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9\} \tag{17}$$

Here s_0 = x, s_1 = y, s_2 = z, s_3 = a, s_4 = b, s_5 = c, s_6 = dt, $s_7 = x_0$ $s_8 = y_0$ and $s_9 = z_0$. Initially s_0= s_7, s_1= s_8 and s_2= s_9

Algorithm 1: The algorithm to generate random whole number less than K using user key/ seed set S:
RandomInt(K, S):

dx = s_3 * (s_1 - s_0) * s_6
dy = ((s_5 - s_2) * s_0 - s_1) * s_6
dz = (s_0 - s_4 * s_2) * s_6
s_0 = s_0 + dx
s_1 = s_1 + dy
s_2 = s_2 + dz
return |([$s_0 * 10^{12}$] ⊕ [$s_1 * 10^{12}$] ⊕ [$s_2 * 10^{12}$])| % K

The square brackets in the above equation represents rounding off operation to its nearest integral value.

Algorithm 2: To generate random double between 0 to 1 using user key S:
Random(S):
dx = s_3 (s_1 - s_0) * s_6
dy = ((s_5 - s_2) * s_0 - s_1) * s_6
dz = (s_0 - s_4 * s_2) * s_6
s_0 = s_0 + dx
s_1 = s_1 + dy
s_2 = s_2 + dz
return $|fractional(s_0 * 10^{12} + s_1 * 10^{12} + s_2 * 10^{12})|$

Therefore $K_c = \{k_{c_0}, k_{c_1}, k_{c_2} ... k_{c_{L*W-1}}\}$ is generated as follows.

for i in range(0, H * W):
k_{c_i} = Random(256, S)

Step 3:To encrypt, every pixel of the image is traversed. Let I_{ij} represent pixel at row i and column j of the image matrix and the encryption sequence be K_c.

Algorithm 3: The algorithm for encrypting the image I using sequence K_c:
Encrypt(I, K_c):
for i in range(0, H):
for j in range(0, W):
$I_{ij} = I_{ij} \oplus k_{c_{i*W+j}}$

Every pixel I_{ij} will be XORed with $k_{c_{i*W+j}}$ i.e (i * W + j)th value the in the sequence K_c. This produces a new encrypted image matrix I_{en_1}.

3.2 Encryption Process With Genetic Algorithm:

The genetic algorithm is an adaptive heuristic search algorithm. It is an soft computing technique which is used to implement artificial intelligence. For a particular problem to be solved we first start with the initial generation of candidate solutions and a function that evaluates how good is the solution based on some fixed parameters called as fitness function. The attributes of the candidate solution which define how the candidate solved the problem is called the chromosome of that solution. Through the stochastic process of G.A., the proposed technique tries to improve these chromosomes. Out of the initial candidate solutions a set of solutions is selected as parents by some stochastic selection process giving greater priority to fitter parents. They are then used to develop a fresh generation of candidate solutions via the operations of crossover and mutation to bring diversity in the solutions. This process is continued and new generations are generated from the previous ones until a stopping condition is reached. The candidate solutions over the generations become more tuned, optimal and better for the given problem.

In the following approach Lorenz chaotic system itself is used to generate all the initial random candidate solutions and introduce the mutations to the selected parents in the heuristic search process of Genetic Algorithm. This therefore helps us surf and eventually discover the locations in the search space where the solution is reasonably optimized i.e. the key sequence gives considerably enhanced entropy/noise after encrypting the image.

In current phase the proposed technique generates another encryption sequence $k_{g_0}, k_{g_1}, k_{g_2}, k_{g_3} ..., k_{g_n}$ and genetic key A. Every value k_{g_i} in K_g is an integer ranging from 0 to 255 and the length of the sequence is equal to the number of pixels in the image I. This sequence is called as genetic encryption sequence K_g. The purpose of K_g is to increase the noise in the image I_{en_1} resulting from the chaotic encryption process and also add another layer of security to the image so that the encryption is stochastic.

Step 1: First, an initial population that is a set of 20 such random sequences are spawned $K^0_{g_0}, K^0_{g_1}, K^0_{g_2}, K^0_{g_3}, ..., K^0_{g_{19}}$ and each of this sequence is generated by Lorenz system similarly to how K_c was generated previously thus each sequence $K^0_{g_i}$ uses a set of Lorenz seed values. Let the seed set corresponding to the ith sequence be denoted as $S^0_{g_i}$ where

$$S^0_{g_i} = \{x_0, y_0, z_0, a + a^0_{g_i}, b + b^0_{g_i}, c + c^0_{g_i}, dt + t^0_{g_i}\} \quad (18)$$

For every set $S^0_{g_i}$ initial Lorenz's state values x_0, y_0 and z_0 remains the same as used by the user in the Lorenz encryption process, only the sampling gap or time step dt and control parameters a, b and c differs by a value of $t^0_{g_i}$, $a^0_{g_i}$, $b^0_{g_i}$, $c^0_{g_i}$ respectively. They are generated randomly such that $10^{-14} \leq t^0_{g_i}, a^0_{g_i}, b^0_{g_i}, c^0_{g_i} \leq 10^{-6}$.
The superscript 0 represents the generation, in this case initial generation. Thus we generate 20 sets $S^0_{g_0}$, $S^0_{g_1}$, $S^0_{g_2}$, $S^0_{g_3}$,...,$S^0_{g_{19}}$ using which we generate 20 keys $K^0_{g_0}, K^0_{g_1}, K^0_{g_2}, K^0_{g_3}, ..., K^0_{g_{19}}$. Now with respect to each sequence $K^0_{g_i}$ of the 20 new sequences we encrypt the image I_{en_1} to generate their corresponding encrypted image $I^0_{g_i}$.
Algorithm 4: The algorithm to obtain the encrypted image using original image I of size H * W and encryption sequence K = $\{k_0, k_1, K_2, ..., k_{H*W}\}$is as follows:
Step 2: Encrypt(I, k):
for i in range(0, H):
for j in range(0, W):
$I_{ij} = I_{ij} \oplus k_{i*W+j}$

Step 3: For the resulting image from each sequence it's respective fitness is calculated as per the following

function:

$$I_{fitness} = I_{entropy} = \sum_{i=0}^{255} p(i) * log_2(1/p(i)) \quad (19)$$

here, in equation 19, $I_{fitness}$ is the fitness the resulting image equal to the $I_{entropy}$ i.e. the entropy of the resulting image. p(i) in equation 19 is the probability of a shade i such that $0 \leq i \leq 255$ in the resulting image given as:

$$p(i) = \frac{freq(i)}{H * W} \quad (20)$$

$freq(i)$ is the frequency or the number of times the shade i occurs in the image I. The greater the value of $I_{entropy}$ the more even is the frequency distribution of the shades in the image and more the noise. For maximum entropy the value freq(i) of every shade i should be:

$$freq(i) \approx \frac{L * W}{256} \quad (21)$$

This state of maxi-ma in the final encrypted image I_{en_2} is what the proposed system tries to achieve via the heuristic search process of genetic algorithm.

Step 4: Of these 20 sequences $K^0_{g_0}, K^0_{g_1}, K^0_{g_2}, K^0_{g_3}, ..., K^0_{g_{19}}$ The parent is selected via Elitism selection process i.e. the parent giving best fitness and a new generation of 20 sequences via mutation process is spawned. Lets say $K^0_{g_i}$ is the selected parent. Values $t^0_{g_i} * 10^{12}$, $a^0_{g_i} * 10^{12}$, $b^0_{g_i} * 10^{12}$ and $c^0_{g_i} * 10^{12}$ are appended to the solution array/ Genetic Key A and the value of it's fitness is stored in global best fitness G and a variable g (generations computed) is set to one i.e. g = 1.

Step 5: After initial population generation and parent selection, mutation is performed in the rest of the generations. In the mutation process first a mutation rate lets say m is set and 20 Lorenz seed sets $S^j_{g_0}, S^j_{g_1}, S^j_{g_2}, S^j_{g_3}, ..., S^j_{g_{19}}$ are generated where:

$$S^j_{g_i} = \{x_0, y_0, z_0, a, b, c, dt + t^j_{g_i}\} \quad (22)$$

Here in above set equation 22 j represents the current iteration or the generation we are on and $t^j_{g_i}$ is a small randomly generated number added to dt of set $S^j_{g_i}$ such that $10^{-14} \leq t^j_{g_i} \leq 10^{-6}$. The values of x_0, y_0, z_0, a, b, c and dt are the same values used in the Lorenz encryption process in user key or seed set. Now suppose the generation is j and selected best parent from the previous generation is K^{j-1}_g.

Algorithm 5: Mutation of parent encryption sequence $K^{j-1}_{g_i}$ to generate i^{th} candidate sequence $K^j_{g_i}$ for j^{th} generation and mutation rate m is as follows:

```
Mutation( K^j_{g_i}, K^{j-1}_{g_i}, t^j_{g_i}, m ):
    S^j_{g_i}={x_0, y_0, z_0, a, b, c, dt+t^j_{g_i}}
    for k in range(0, H * W):
        r = Random(S^j_{g_i}):
        if r ≤ m :
            K^j_{g_i}[k] = K^{j-1}_{g_i}[k] + RandomInt(256, S^j_{g_i})
        else:
            K^j_{g_i}[k] = K^{j-1}_{g_i}[k]
```

Here $K^{j-1}_{g_i}[k]$ and $K^j_{g_i}[k]$ represents the k^{th} value in the sequence $K^{j-1}_{g_i}$ and $K^j_{g_i}$ respectively.

Step 6: Thus 20 independent mutations are performed on selected parent sequence K^{j-1}_g to generate 20 separate candidate sequences $K^j_{g_0}, K^j_{g_1}, K^j_{g_2}, K^j_{g_3}, ..., K^j_{g_{19}}$ and with respect to each the image I_{en_1} is encrypted using the Encrypt algorithm shown before to generate $I^j_{g_0}, I^j_{g_1}, I^j_{g_2}, ..., I^j_{g_{19}}$ respectively. Calculate the fitness of each of these images and the sequence which gives the image with fitness greater than G after encryption is selected via elitism selection process, lets say $K^j_{g_i}$, this is passed on to the next generation as parent and its corresponding value of $t^j_{g_i} * 10^{12}$ is appended to the solution array A. Update G as fitness of image $I^j_{g_i}$. If none of the child candidates in the current generation are able to perform better than current G, the entire generation is discarded and the process is repeated. For every computed generation set g = g + 1. These operations are performed unless a stopping condition is reached. The stopping condition being that G ≤ 7.9997 or g ≤ 200 whichever comes first. Finally a solution array A = $a_0, a_1, a_2, ..., a_l$ is generated which is our genetic key and will be an array of values where $3 \leq l \leq 203$ and the image with entropy G is our final encrypted image I_{en_2}.

3.3 Decryption Process:

The decryption process consists of the following steps.

Step 1: The chaotic encryption sequence $K_c = \{k_{c_0}, k_{c_1}, k_{c_2}, k_{c_3}, ..., k_{c_n}\}$ is regenerated using the values a, b, c, x_0, y_0, z_0 and dt that are entered by the user that is the user key.

Step 2: For genetic decryption we first generate the initial sequence K form the Lorenz seed set S_0 derived from the genetic key A as follows

$$S_0 = \{x_0, y_0, z_0, a+A[0], b+A[1], c+A[2], dt+A[3]\} \quad (23)$$

Step 3: Then perform mutation on the key sequence K:
Algorithm 6: Mutation on key sequence K using seed set S and genetic key A and mutation rate m:

```
Mutation( S, A, K, m ):
    for j in range(4, l + 1):
        S_{j-3}={x_0, y_0, z_0, a, b, c, dt + A[j]}
        for i in range(0, H * W):
            r := Random(S_{j-3})
            if r ≤ m:
                K[i] = K[i] ⊕ RandomInt(256, S_{j-3})
```

Thus K is mutated into the genetic encryption sequence K_g

Step 4: Using the generated keys in the above process
$K_g = \{k_{g_0}, k_{g_1}, k_{g_2}, ..., k_{g_n}\}$ and $K_c = \{k_{c_0}, k_{c_1}, k_{c_2}, ..., k_{c_n}\}$ to

decrypt image I_{en_2}

Algorithm 7: Decryption of image I_{en_2} using encryption sequences K_g and K_c is as follows:
Decrypt(I_{en2}, K_g, K_c):
for m in range(0, H):
for n in range(0, W):
$I_{mn} := I_{mn} \oplus k_{g_{m*W+n}} \oplus k_{c_{m*W+n}}$

Where I_{mn} represents pixel at position (m,n) of image I_{en_2}. This will thus help us the original image I. Once the Genetic key A for the image is obtained, it can be used for both future encryption and decryption rapidly.

4 Results And Discussion

This chapter provides us with the detailed analysis and outcome of the proposed image encryption technique in terms of image entropy, standard deviation of pixel states and time. It also analyses the Lorenz sequence sensitivity with respect to the initial seed values(user's key).

4.1 Results

Through the proposed work the following are the outcomes.
• The first stage involves using Lorenz chaotic system which generates a sequence K_c XORed onto the original image figure 1 Cameraman of size 256x256. Applying the

Figure 1. Original Image

Chaotic encryption process we get the encrypted figure 2. This is the intermediate encrypted image denoted as I_{en_1} in the proposed methodology. I_{en_1} has an image Entropy of 7.99691 and the control parameters a, b and c, initially x, y, and z state values i.e.: x_0, y_0 and z_0 and sampling gap/step size dt in they user key are as follows

a = 14.996842793294382, b = 27.999174693379107, c = 2.656926910693863,
x_0 = 21.991698467920756, y_0 = 11.993968326510558,
z_0 = 8.998217020214971,
dt = 0.010007020214971

• The part following this is the generation of genetic key and encryption sequence to add the second layer of security which adds stochastic elements to the encryption

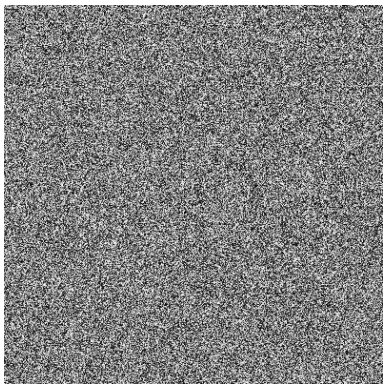

Figure 2. Encrypted Image after first stage

process and enhances the image noise. Here new Lorenz seeds sets are generated from seed values used in the previous process to generate random initial population of candidate solutions and the best are selected by Elitism selection. Then again new seed sets are used to generate random sequence to mutate the parent and next generation is generated. Mutation rate of 0.5% i.e. m = 0.005 was chosen. Described operations are carried out for over 200 iterations or till the an image with entropy greater than 7.99971 is reached which ever comes first. Thus out our final encryption sequence K_g, genetic key A and final encrypted image I_{en_2} figure 3 is obtained.

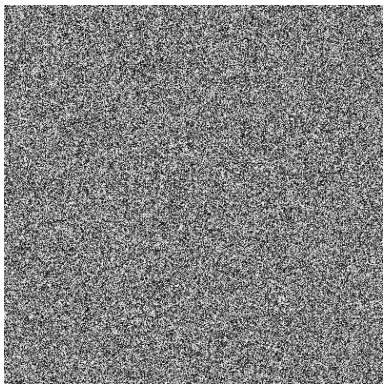

Figure 3. Encrypted Image after final stage

The figure 4 demonstrates the gradual improvement in the encrypted image fitness as the whole G.A. process progresses. Chaotic encryption gave an image noise of 7.99691 which was improved by genetic Key to 7.9997. All of these methods have image entropy greater than 7.9997 which means information leakage in the encryption scheme is negligible.
• Now if we change the value of dt during the decryption process by adding 10^{-14} and then try to decrypt the image:

Thus the proposed system has demonstrated the high sensitivity of the user key and decryption process to the initial parameters. Changing even one of the parameters by a factor of 10^{-14} will lead to a completely different sequence hence failing to successfully decrypt the image. This is because Lyapunov exponents are a measure that is

```
Fitness: 7.99944        Generation: 4
Fitness: 7.99945        Generation: 5
Fitness: 7.99946        Generation: 6
Fitness: 7.99947        Generation: 7
Fitness: 7.99948        Generation: 8
Fitness: 7.99950        Generation: 9
Fitness: 7.99950        Generation: 10
Fitness: 7.99951        Generation: 11
Fitness: 7.99953        Generation: 12
Fitness: 7.99954        Generation: 13
Fitness: 7.99955        Generation: 14
Fitness: 7.99956        Generation: 15
Fitness: 7.99957        Generation: 16
Fitness: 7.99957        Generation: 17
Fitness: 7.99959        Generation: 18
Fitness: 7.99960        Generation: 19
Fitness: 7.99961        Generation: 20
Fitness: 7.99961        Generation: 21
Fitness: 7.99962        Generation: 22
Fitness: 7.99962        Generation: 23
Fitness: 7.99963        Generation: 24
Fitness: 7.99963        Generation: 25
Fitness: 7.99964        Generation: 26
Fitness: 7.99965        Generation: 27
Fitness: 7.99965        Generation: 28
Fitness: 7.99967        Generation: 29
Fitness: 7.99967        Generation: 30
Fitness: 7.99967        Generation: 31
Fitness: 7.99968        Generation: 32
Fitness: 7.99969        Generation: 33
Fitness: 7.99969        Generation: 34
```

Figure 4. Genetic Optimization

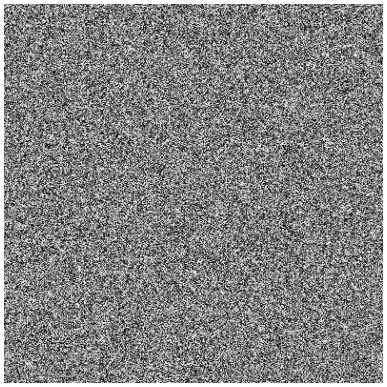

Figure 5. Failed decryption using wrong key

used to quantify the sensitivity of system to initial state or condition. Such chaotic systems or Lorenz system in particular that exhibit chaos have positive Lyapunov exponents.

4.2 Evaluation Parameters

The evaluation parameters taken into consideration for analysis are as follows:
• Key Space Analysis: In order to review the proposed image encryption design, the key analysis is very essential. In this analysis, the key space is required to be sufficiently large to make the brute-force attack unattainable.

• Histogram Analysis: Histogram analysis shows the unique distribution of plain image. For good encryption design, encrypted image should attempt to erase the traces of the plain image. In order to safeguard the information of the original image, it is imperative for the encrypted image to exhibit no statistical similarity with the original image.

• Entropy Analysis: The entropy analysis is practiced to assess the randomness of an image. If an image has an excellent random property, it determines that the entropy score is close to the maximum entropy value. Its value can be computed by the following equation:

$$I_{entropy} = \sum_{i=0}^{255} p(i) * log_2(1/p(i)) \tag{24}$$

where p(i) is the probability of a shade i such that $0 \leq i \leq 255$ in the resulting image. The greater the value of $I_{entropy}$ the more even is the frequency distribution of the shades in the image and more the noise. For maximum entropy the value p(i) of every shade i should be:

$$p(i) \approx \frac{L * W}{256} \tag{25}$$

• Correlation Coefficient Analysis: The security level of cryptography system is also commonly measured in terms of diffusion and confusion. Therefore, statistical analysis is conducted on the image by testing the correlation 10,000 randomly selected pairs of two adjacent pixels in the original image and encrypted image. Next, the correlation of each pair is calculated using the following equations:

$$r_{xy} = \frac{cov(x,y)}{(\sqrt{D(x)})(\sqrt{D(y)})} \tag{26}$$

where,

$$cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - (\frac{1}{N}\sum_{i=1}^{N}x_i))(y_i - (\frac{1}{N}\sum_{i=1}^{N}y_i)) \tag{27}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - (\frac{1}{N}\sum_{i=1}^{N}x_i))^2 \tag{28}$$

Where x_i and y_i are random selected pair pixels.

4.3 Key Space Analysis

In the proposed algorithm, the initial conditions of the Lorenz Chaotic System have been employed as the secret keys with the precision of 10^{-14}.
An efficient chaotic system is distinguished for its high sensitivity to the initial conditions. In the proposed algorithm, the initial control parameters 'a', 'b', 'c' and the sampling interval 'dt' each of 14 digit precision post-decimal points is the user secret key or password. The key length is 56 digit, hence 56! i.e (7.1099859e + 74) combinations are possible. The key space of our proposed system is 10^{56}. Ideally, the key space should be more than 2^{100} to provide adequate security in order to evade brute-force attacks. According to this, the key space is large enough for the secret keys and to withstand all kinds of brute-force attacks.
In the proposed algorithm, the initial control parameters 'a', 'b', 'c' and the sampling interval 'dt' each of 14 digit precision post-decimal point is the user secret key or password.

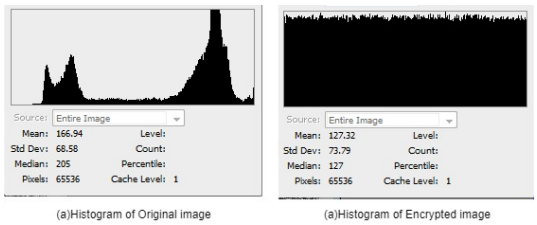

Figure 6. Histograms of the Lena image (a) Original images (b) Encrypted image

4.4 Histogram Analysis

Figure 6 shows the histogram of the original image Lena with the size of 256 x 256 and its corresponding encrypted image. It is observed that histogram of encrypted image is very uniform, and the regularity of plain image is not bought into the encrypted image. The domain of the histogram are discrete integers ranging from 0 to 255 representing pixel shades. The Y axis also contains integers representing the number of occurrences.

4.5 Entropy Analysis

The below Table I shows the entropy of two different images Lena and Cameraman.The table addresses the entropy value for the plain input image and encrypted image. Lena and Cameraman image are of size 256x256

Table 1. Entropy Of Images

Images	Lena	Camera Man
Plain Image	7.575	7.0710
Encrypted Image	7.99965	7.9997

4.6 Correlation Coefficient Analysis

The correlation coefficient between adjacent pixels of original image and encrypted image are given in Table II. The experimental results show that the correlation coefficient between adjacent pixels are high in plain original image and being greatly weakened in the encrypted image Ref [14].The proposed encryption scheme is perfect for coefficient analysis. Lena and Cameraman image are of size 256x256

Table 2. Correlation Coefficient Of Cameraman Image after and before encryption

Criterion	Lena	Cameraman
Input Image	0.9268	0.9567
Encrypted Image	0.00548	0.00145

The comparison of correlation coefficient and entropy of proposed method with Cameraman image 256*256 with other methods is shown in the below table 3.

It is observed in the table above that the entropy and correlation coefficient of proposed method is better than the referred methods. In figure 7 we can see that the disorder brought about in the encrypted image by the proposed method is more than that brought about by the individual methods. Similarly figure 8 shows that the encrypted image produced by the proposed method has least correlation to the original image in comparison to the other methods.

Table 3. Correlation Coefficient Of Cameraman Image after and before encryption

Method	Correlation	Entropy
Lorenz Chaotic System Ref[14]	0.0054	7.9971
Genetic Algorithm Ref[12]	0.0048	7.9904

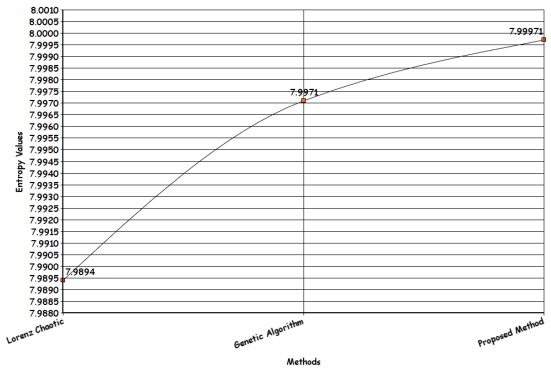

Figure 7. Entropy analysis of Cameraman Image

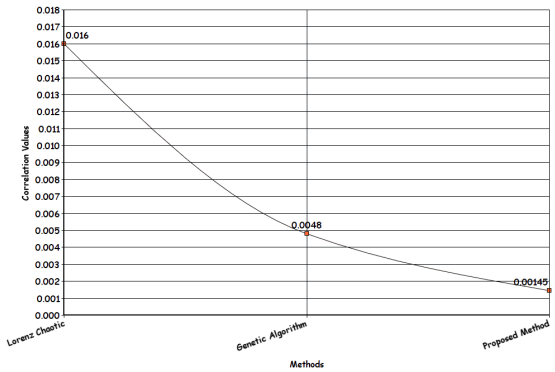

Figure 8. Correlation analysis of Cameraman Image

5 Conclusion

Proposed system has shown an easy and highly optimized way of encrypting images which has been clarified by our design and methodology. Technique first utilizes Lorenz chaotic system of differential equation to generate a random sequence of integers ranging from 0 to 255 which are highly dependent on the initial values, time step and control parameters which form a key that generates sequence K_c to achieve the first level of encryption and a noisy image. Given that the whole sequence can be generated using just the initial parameters the whole sequences need not be stored apart from those initial parameters that are the seed values making it space efficient. Also these values should be accurate up to 10^{-14} decimal place ensuring high key

sensitivity, changing those values even slightly will give a completely different image during decryption. Following Lorenz encryption, Genetic algorithm used hence is used to optimize the noise in the encrypted image I_{en_1} generated by the Lorenz System even further making it more random and chaotic giving us a second layer of security to nullify the deterministic nature of Lorenz system and an image entropy close to or over 7.9997 with adequately small pixel correlation of the final encrypted image I_{en_2}. We have hereby demonstrated a way to use Lorenz system to generate the candidate solutions and introduce random mutations in the heuristic search process of Genetic Algorithm to eventually locate the optimum solution in the search space. The merit of our proposed system lies in the fact that our genetic sequence K_g i.e. the solution once generated can be regenerated easily by the genetic key A of length 200 or less for both encryption and decryption process, thus is space efficient because only this array needs to be stored and not the entire sequence.

References

[1] Ruisong Ye, Yinhua Li, Yajuan Li *An Image Encryption Scheme Based on Fractal Interpolation ACM 2018.*.Department of Mathematics, Shantou University Shantou, Guangdong, 515063, China e-mail: rsye@stu.edu.cn

[2] Xingyuan Wamg , Xiaoqiang Zhu , and Yingqian Zhang *"An Image Encryption Algorithm Based on Josephus Traversing and Mixed Chaotic Map" IEEE 2018...School of Information Science and Technology, Dalian Maritime University, Dalian 116026*

[3] Ch. K. Volos, I. M. Kyprianidis, I. N. Stouboulos *"Image encryption process based on a Chaotic True Random Bit Generator" IEEE 2009...Department of Physics, Aristotle University of Thessaloniki*

[4] Saswat Pujari, Gargi Bhattacharjee, Sumyakanta Bhoi *"Hybridized Model For Image Encryption Through Genetic Algorithm And DNA Sequence"*...Tata Consultancy Mumbai, Department of Information Technology, Veer Surendra Sai University of Technology, Burla, Odisha, India Elsevier-2018.

[5] Obaida M. Al-Hazaimeh1, Mohammad F. Al-Jamal2, Nouh Alhindawi3, Abedalkareem Omari *"Image encryption algorithm based on Lorenz chaotic map with dynamic secret keys" SPRINGER 2017*...The Natural Computing Applications Forum 2017

[6] P.Rajkumar, V.Balamurgan, Priyanka Mishra *"Adaptive Genetic Algorithm for a Real Time Medical Images"*...Faculty of Electrical and Electronics Sathyabama university Sathyabama University Chennai-600119... IEEE 2016

[7] Shikha Pathak, Vikas Sejwar *"Optimized Noisy Image Segmentation Using Genetic Algorithm"*Department of CSE and IT Madhav Institute of Technology and Science Gwalior, India...IEEE 2017.

[8] M. I. YOUSSEF, M. ZAHARA, A. E. EMAM, and M. ABD ELGHANY *"Image Encryption Using Pseudo Random Number and Chaotic Sequence Generators"*Al-Azhar University - Cairo - Egypt.

[9] Rafik Hamza *"A novel pseudo random sequence generator for image-cryptographic applications"*LAMIE Laboratory, Department of Computer Science, University of Batna 2, Batna, Algeria...Elsevier 2017

[10] Xing-YuanWang · Lei Yang · Rong Liu · *"A chaotic image encryption algorithm based on perceptron model"*Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China...Springer 2010

[11] Ashwin Ramesh,Akash Jain *"Hybrid Image Encryption using Pseudo Random Number Generators, and Transposition and Substitution Techniques"*Department of CSE MIT, Manipal...IEEE 2015

[12] Saeed Mozaffari· *"Parallel image encryption with bitplane decomposition and genetic algorithm"*Faculty of Electrical and Computer Engineering, Semnan University, Semnan, Iran...Springer 2018

[13] Mohamed A. Mokhtar1, Nayra M.Sadek2, Amira G. Mohamed3· *"Design of Image Encryption Algorithm Based on Different Chaotic Mapping"* Faculty of Engineering, Alexandria University, Alexandria, Egypt...IEEE 2017

[14] Obaida M. Al-Hazaimeh, Mohammad F. Al-Jamal, Nouh Alhindawi, Abedalkareem Omari· *"Image encryption algorithm based on Lorenz chaotic map with dynamic secret keys"*Department of Computer Science, Al-Balqa' Applied University, Al-Huson University College, P. O. Box 50, Al-Huson, Irbid, Jordan...Springer 2017