ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

IMAGE ENCRYPTION USING LOGISTIC-COSINE-SINE CHAOS MAP AND ELLIPTIC CURVE CRYPTOGRAPHY

BHAT JASRA[1], MANASHA SAQIB[2], AYAZ HASSAN MOON[2]

[1,2,3] School Of Engineering and Technology, IUST Awantipora, J&K-192122

{[1]jasra.bhat, [2]manasha,saqib, [3]ayaz.moon}@islamicuniversity.edu.in

ABSTRACT

A number of image encryption schemes based upon chaotic systems proposed in the past, have been found to be vulnerable to known-plaintext attack due to the usage of simple linear mapping functions, predictable trajectories etc. In this paper, we propose a hybrid scheme based upon Logistic-Cosine-Sine chaos system and Elliptic Curve Cryptography. The complex chaos mapping function shall result in uniformly distributed pseudo-random outputs and a wider dynamic range while as the Elliptic Curve based cryptography shall be used to leverage the computationally secure encryption of the image pixels. Simulation of the proposed algorithm has been carried using 512-bit standard Elliptic curve given by ECC Brainpool. Results, security analysis reveal that the proposed encryption technique is both computationally efficient as well as resilient against different types of attacks. Comparative analysis shows that the proposed scheme uses lesser number of point multiplications for encryption, and is faster as compared to similar previous schemes.

Keywords: *Image Encryption; Image security, Elliptical Curve Cryptography (ECC), Image Encryption Using ECC, Image Encryption Using Chaos theory;*

1. INTRODUCTION

Information exchange using images and other multimedia data has grown exponentially. This has been fuelled by the availability of a wider range of networking platforms like social networking sites, affordability of devices like smartphones, laptops, HD cameras and the accessibility to very high-speed internet connectivity. While sharing images over internet is rampant, the potential security risks associated with it have also increased. Important security primitives like confidentiality and authentication associated with the transmission and distribution of sensitive images over public networks is therefore assuming critical importance**.** In many application scenarios, images need to be protected from potential security threats like unauthorized modification, copyright violations, prohibited access to view or modify the contents of images. Just like for normal textual data, lot of techniques have been proposed by researchers for ensuring secure transmission of images**.** Images are usually larger in size, even a small greyscale image of 256*256 contains a set of 65536 pixels, have very high inter-pixel correlation and are multidimensional. In general, the data redundancy within an image is very high. All these characteristics of images make the conventional text encryption techniques infeasible for their applicability to images.

In spatial domain most commonly used encryption approaches are based on pixel permutation or substitution. Chaos map theory is one of the most widely used permutation-based technique for image encryption [1]. A variety of chaotic systems (logical map, tent map, Arnold map [2] etc) have been used by researchers. Every Chaos system uses a complex mapping function which maps any input sequence on to a Pseudo random sequence [3][4][5][6]. The randomness/ efficiency of this mapping is dependent on the complexity of the mapping function. The output is highly sensitive to changes in initial control parameters of the system. The initial conditions and other control parameters that act as seed for the chaos system, are analogous to secret key used in cryptosystems. Chaos systems can be used in many ways for encrypting image such as:

i) It can be used to generate a pseudo random sequence that can be used as a key in any conventional cryptosystem [7].

ii) The output of a chaos system can be used to scramble pixel positions in original image and repeated iterations of such scrambling produce an encrypted image [8].

Researchers have followed different approaches to secure images and other multimedia data. These include watermarking, image steganography, conventional symmetric key encryption techniques like DES, T-DES, Hill cipher, public key encryption techniques like RSA, AES [9-11], Chaos theory-based encryption/pixel scrambling techniques, and Digital signature-based techniques etc. Among all these ECC and chaos maps are used extensively in recent past, either in combination or separately. However, it has been observed that using simple chaotic maps alone to encrypt images is vulnerable to many security attacks such as known-plaintext attack etc. If the mapping function used is not complex enough, the chaotic behaviour becomes easily predictable and tends to stable at smaller key space, hence reducing the security strength of the system. In order to overcome these gaps researchers have often used Chaos systems in combination with other cryptosystems like DNA computing [12], Hill Cipher, ECC etc or use more than one simpler chaotic functions together to form better, secure systems with increasing ranges and dynamics.

Elliptic Curve Cryptography (ECC) [13], a public key-based Cryptosystem (PKE) is one of the choicest techniques used for attaining encryption, authentication or key exchange for which it leverages hardness of Elliptical Curve Discrete Logarithmic Problem (ECDLP). ECC has been immensely used to secure images [14,15,16]; Image secrecy is obtained by using Elgemal ECC encryption, Image authentication by using ECC based digital signature schemes. ECC based Key exchange schemes are used to aid other image encryption techniques like chaos maps, AES, DES etc.

In our technique we will exhaust both ECC based key exchange and encryption technique along with Logistic-Cosine-Sine based chaos system to attain image secrecy in a simpler yet efficiently secure manner.

Kamrani A. and others use Logistic chaos map along with discrete orthogonal moment functions to encrypt images [17]. A pre-operated 128bit key is used to generate control parameters of the chaos system. Initially the plain image is represented using discrete moments transform, followed by n iterations of confusion and diffusion of pixel values using logistic chaos map. The proposed method outperforms similar techniques like DCT etc. N. Sasikaladevi, K. Geetha and others propose a three-stage encryption process [18]. In stage one keyed logistic map is used to generate two chaotic sequences for every colour plane, in second stage DNA encoding/Scrambling and rotation is done, and finally every pixel pair of DNA encoded image is encrypted using ECC. Using three layered masking helps in attaining better resistance against attacks. M.A.B. Farah, et al. also proposed a multistage encryption method based on chaos, DNA encoding and fractional Fourier transform [19]. The input image is subject to DNA encoding, and then XORed with hashed input image and a chaotic matrix is obtained using Lorenz mapping function. The output is then subjected to fractional Fourier transform thrice, to get the final cipher image. Experimental results show resistance of proposed scheme to most attacks. Singh and Singh used Logistic, Arnold chaos mapping functions to encrypt a plain image. Elliptic Curve Diffie Hellman key exchange is used to establish a secret key between sender and receiver [2]. This key is then used to generate the control parameters of the logistic chaos system (LCS). A Chaos sequence is generated using LCS, which after converting to integer values is multiplied with shared curve point to produce a sequence of curve points. These points are then converted into byte sequences which are then XORed with scrambled plain image pixels (using Arnold's map) to get the cipher image. X. Zhang and A.X. Whang used ECC along with Piece-Wise linear chaotic map to develop an image encryption technique [20]. Initially a key is generated by performing SHA-256 on plain image and shared using ECDH. This key is used to generate a chaotic sequence using Piece-Wise linear chaotic map. The pixels of plain image are base converted to large integers of 64 bits and then encrypted using ECC. This ECC encrypted image is then XORed with the already obtained random chaotic sequence to get the encrypted image. This method efficiently reduces the number of ECC operations needed to encrypt the image. Ziad E. Dawahdeh and team used ECC to establish an initial key between sender and receiver, this key is then used to generate a self-invertible key matrix at both parties. Using this key matrix in standard Hill-Cipher algorithm, a plain image is encrypted. At decryption end, the same key matrix is used to recover the original image. This way secrecy of key in hill cipher is improved [21]. In [22] a colour image encryption based on AES and ECC is proposed. Initially a signed ECDH is used for key exchange; A sequence of random numbers is generated using ECC, which is later used along with AES to encrypt the plain image. In [23] SHA-512 is applied on plain image to get the keys for chaos system. Subsequently Elgamal encryption and cross permutation, DNA sequencing is used to generate the cipher image. K. Gupta, S. Silkari in [24] used standard map along with rotations of colour planes of original image to form a diffusion template, 3D-Cat map along with rotations of colour planes is used to achieve shuffling, the cipher image is produced by XORing the diffused and shuffled image. They also used ECC based key exchange to establish key used for generating initial parameters of the chaos systems. Singh and Singh also used Elgamal ECC encryption to secure and authenticate an image [25]. In their technique they used pixel grouping approach, i.e. instead of encrypting individual pixels, a group of pixels is converted into a large number (using base conversion) based on the size of Elliptic group used. Consecutive groups are then taken in pairs and encrypted as points on curve. This approach aims at reducing number of ECC multiplications used in encryption and decryption process. S. Bakhtiari, S. Ibrahim et al. use ECC to selectively encrypt a JPEG image. They take 8*8 blocks of original image, apply DCT over each block and discard higher frequencies; ECC is applied only to lower frequency/ DC coefficients of every block to get encrypted image [26]. A. Soleymani and others proposed a novel technique for mapping pixels to points on an elliptic curve before encryption. All possible points on chosen curve are taken and arranged into 256 groups. A mapping table is created where rows represent intensity values (0 to 255) and columns represents a point on curve. Using this mapping table every pixel can be transformed to and from Elliptic curve points for encryption and decryption [27]. Yadav et al. also mapped pixels of original image over a binary group elliptic curve by encoding each point as an exponent of the generator point. The resultant points are then subject to Elgamel encryption and decoded back to pixel domain to form the cipher image [28]. Gupta and Silkari used koblitz mapping technique where every pixel intensity m of a binary image is transformed onto curve point (x_p,y_p) as:

$$x_p = m * k + j \quad (1)$$

Where j=0,1,2,..M and k is a random positive integer pre-decided between sender and receiver such that m*k<p.

$$y_p = sqrt(x^3 + ax + b) \quad (2)$$

If y_p in Eq.2 corresponding to x_p is an integer then (x_p, y_p) is mapped to m, else we keep on checking until $j = k - 1$. If no integer value for y_p is found in this interval, then the point is not found with the specified value of k.The resultant points are then encrypted and reverse mapped onto pixel domain to get cipher image [29]. Lan et al. used cascading, non-linear combinations and switching operations over simple one-dimensional chaotic functions to produce an integrated chaotic system. Using the resultant map for encrypting images proves more efficient and secure as its dynamic range, randomness and sensitivity is much better than the individual seed chaotic maps [30]. In [3], Logistic chaos map and Cyclic Elliptic Curve based pseudo random number generator is used to obtain a key stream. The parameters of Logistic map are controlled by a secret user key. Encrypted image is obtained by masking the original image with the key stream obtained. Statistical analysis show that this scheme is efficiently secure.

The rest of this paper is organized as follows: In section 2, we will discuss fundamentals of Elliptic curve cryptography and Logistic-sine-cosine chaos system. Section 3 discusses proposed encryption and decryption scheme in detail. Simulation details, Results and Security analysis is given in section 4. We also present a comparative analysis with similar recent works in section 4.

2. MATHEMATICAL PRELIMINARIES

In this section we shall discuss the basic operating principles of the chaos function used i.e. LCS and Elliptic curve cryptography.

2.1 Logistic- Sine- Cosine Chaos System

A number of chaotic systems have been proposed over past few years for image encryption. However, many such systems fail to provide perfect security due to simpler mapping functions, predictable trajectories etc. making them vulnerable to attacks like known plaintext attack. Therefore, it is necessary to use a chaos system whose mapping function is complex enough, produces randomly distributed outputs and a larger key space. In our paper we use Logistic sine cosine chaos system which is a more specific form of Cosine transform based chaos system (CTBCS) that meets the above requirements. As given in Eq. (3) CTBCS can be defined as chaos system that combines two simple chaos systems to form a better one.

$$xi + 1 = \cos(\pi(F1(a, xi) + F2(b, xi) + \alpha)) \qquad (3)$$

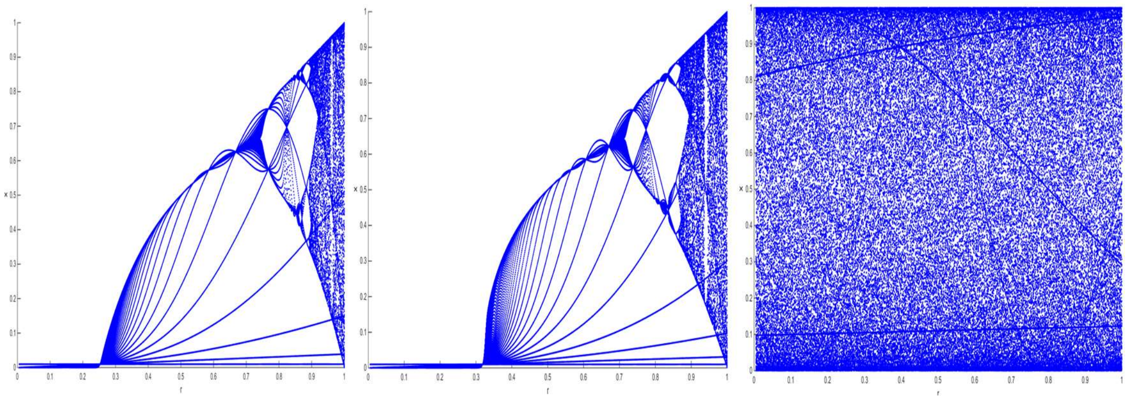
Where $F1(a, xi), F2(b, xi)$ are two individual maps, with control parameters a, b respectively and α is the shifting factor used.

Using Logistic, sine map functions as seed maps in Eqn.3 gives us our logistic-sine-cosine map. i.e.

$$x_{i+1} = \cos(\pi(4rx_i(1 - x_i) + (1 - r)\sin(\pi\, x_i) - \alpha)) \qquad (4)$$

Where r is the control parameter, r ∈ [0, 1].

It can be observed from Eq. (4), the logistic-sine-cosine map takes output of logistic map and sine map, combines them using shifting constant α, and then applies Cosine transform on the combined output. This way the chaotic complexity of the entire system is improved. The chaotic behaviour and hence the security strength of the resultant hybrid system is much more dynamic, random than the individual logistic and sine-based seed maps. The bifurcation diagrams (Figure 1.) clearly show that the dynamic range, chaotic space of proposed chaos map is better and more random than the individual seed maps.



(a) (b) (c)

Figure 1. Bifurcation diagrams of a) Logistic map b) Sine Map c) Logistic-Sine-Cosine Map.

2.2 Elliptic Curve Cryptography

An Elliptic curve unlike the name suggests are not ellipses but are topologically more like Tori defined by a cubic equation in 2 variables:

$$\mathrm{y}^2 = x^3 + \mathrm{ax} + \mathrm{b}$$

Eq. (5) is known as normalized Weierstrass equation is used for cryptography. The values of coefficients and variables are restricted to elements of a finite field. Usually, two types of curves are used for cryptographic purposes: Curves over prime field and curves over binary field.

2.2.1 Elliptical curve defined over prime field Fp

Elliptical curve over a prime field Fp is defined as:

$$\mathrm{y}^2 \mathrm{mod\ p} = (x^3 + \mathrm{ax} + \mathrm{b})\mathrm{mod\ p} \qquad (6)$$

Eq. (6) is symmetric over y=0 and the elements in finite field vary from 0 to p-1. It must satisfy the constraint $4a^3 + 27b^2 \bmod p \neq 0$, to avoid occurrence of repeating roots that makes it feasible for cryptographic use.

The prime number p must be taken in such a way that the number of points $\#E_{Fp}$ is sufficiently large. As

$$p + 1 - 2\sqrt{p} \le E_{F_P} \le p + 1 + 2\sqrt{p} \qquad (7)$$

Standard for efficient cryptography (SEC) recommends p to be 112 to 512 bits long for good security.

The domain parameters for Elliptical curve over Fp are: prime number p, coefficients a and b, generator point G on curve, the smallest possible integer n known as order of G such that for cyclic group n*G=O; where O is known as zero point or point at infinity.

2.2.2 Operations in Elliptic Curve Cryptography

An Elliptical cryptosystem consist of a set of domain parameters and operations over the curve. These operations form the basis of cryptography using elliptic curve.

1. Point addition:

Let $A(x_a, y_a), B(x_b, y_b)$ be two points on the curve defined by Eqn. 3.Then $C(x_c, y_c)$ such that C=$A + B$ is defined as:

$$x_c = s^2 - x_a - x_b \qquad (8)$$

$$y_c = -y_a + s(x_a - x_c) \qquad (9)$$

$$s = \frac{y_a - y_b}{(x_a - x_{b)}} \qquad (10)$$

Where s is the slope of line passing through A and B.

Subtracting point B from A is equivalent to adding mirror image of B over y axis to A i.e.

$$A(x_a, y_a) - B(x_b, y_b) = A(x_a, y_a) + B(x_b, -y_b) \qquad (11)$$

*(Note that the Abelian groups over which elliptical curves are defined have closure over addition and multiplication) i.e. for any binary operation *, if A, B belong to group then C=A*B also is in the group.*

2. Point Doubling:

To add a point A (x_a,y_a) to itself on elliptical curve is called point doubling. Let $B(x_b, y_b) = 2A$

$$x_b = ((3x_a^2 + a)/2y_a)^2 - 2x_a \qquad (12)$$

$$y_b = (\left(\frac{3x_a^2 + a}{2y_a}\right)(x_a - x_b) - y_b \qquad (13)$$

Geometrically it can be achieved by drawing a tangent on curve through A, and the mirror image on y-axis of the intersection point of this tangent on curve is double of A.

3. Point Multiplication:

Multiplication on Elliptical curve is achieved by repeated addition. Let S be a scalar then $S * A$ means adding A to itself s times. It can be achieved using point doubling and point addition operations. E.g., $3A = A + A + A$ i.e., $2A + A$.

2.2.3. Encryption and Decryption Using Elliptic Curves

Let $\mathrm{y}^2 \mathrm{mod\ p} = (x^3 + \mathrm{ax} + \mathrm{b})\mathrm{mod\ p}$ be an elliptical curve over prime field with a, b, p, G, n as domain parameters. Let M be the message to be encrypted. Let A_m be the point on curve to which M is mapped.The sender selects a random integer n_A as its private key and calculates its public key $P_A = n_A * G$ using point multiplication on ECC.The receiver selects n_B a random integer as its private key and public key $P_B = n_B * G$.

Both sender and receiver share their public keys with each other. The Encrypted message C_m is a pair of points on curve as:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$C_m = (n_A * G, A_M + n_A p_B) \quad (14)$$

At Decryption end the receiver multiplies its private key with first point i.e. $n_B * n_A * G$ and recovers encrypted point by subtracting it from 2nd point of C_m i.e.

$$A_M = (A_M + n_A * P_B - n_A * n_B * G) \quad (15)$$

3. PROPOSED SCHEME

In the proposed encryption scheme, a secret key is established between sender and receiver. This key is used to generate the control parameters of logistic-cosine-sine chaos system. A pseudo random sequence of size equal to plain image is generated using this chaos system. The pixels in plain image are grouped to form large integers based on the length of key of the elliptic curve used. We have used complementary function *FromDigits* and *IntegerDigits* of Wolfram Mathematica version 12 to group pixels based on base conversion. These large integers are then encrypted using ECC. The encrypted integers are converted back into pixel values which are then XORed with the already generated Pseudo random chaos sequence to obtain the cipher image. This scheme is simple and elegant and can be used with varying key size. The steps involved are described below:

3.1 Key Exchange

Let Alice be the sender and Bob, the receiver. Both Alice and Bob must agree on an Elliptic curve $EC(F_P)$ with generator point G, prime field F_p; Assuming both Alice and Bob know each other's public keys. Let n_A , P_A be the private and public keys of Alice respectively. Let n_B ,P_B be the private and public keys of Bob respectively. The steps involved in the process of key exchange are as following:

- For plain image I, $h = SHA512(I)$.
- Alice calculates $M = (n_A + h) * P_B$ and sends it to Bob;
- Bob receives M and extracts the key $K1$ as:

$$K1 = M1 * (n_B)^{-1} - P_A \quad (16)$$

- Point $K1 = h * G$ is the key established between Alice and Bob. Here * denotes multiplication over Elliptic curve.

3.2 Generating Pseudo Random Number Sequence Using Logistic-Cosine-Sine Map.

- Using key $k1 = (x_{k1}, y_{k1})$, Set initial parameters as $x_0 = x_{k1}/p$; $r = y_{k1}/p$.
- Using x_0, r in Eq. (4), generate a pseudo random chaos sequence CS equal to size of plain image I.
  If computing precision of machine is 10^{16}, get an integer sequence of range 0 to 255 using:
  $$y[i] = Round\big[Mod[CS[i] * 10^{16}, 255]\big] \quad (17)$$
- Reshape this sequence to form a random chaos image I_A, equivalent to size of original image.

3.3 Encrypting Original Image

Given an original image I of size $m \times n$. Encrypted image is obtained as follows:

- Alice chooses a random secret number x such that $x < n$
- Alice calculates $K2 = x * P_B$, $xG = x * G$,
- Flatten the image I into $1D$ array of pixels.
- Add Random 1 or 2 to every pixel to account for errors incurred while grouping and reverse grouping.
- Using *FromDigits* function, Group consecutive pixels to form big integers in base 258. Where length of group is given by $Length[Integer\ Digits[p, 258]] - 1$; For a 512bit curve 63 pixels can be grouped together.
- Pair- up the consecutive big integers obtained above to form Points (P_m).
- For every point P_m perform point addition to get $P_c = P_m + K2$;
- Using *IntegerDigits* get convert P_C into integers of range 0 to 255.
- Reshape above sequence to get image I_B equivalent to size of original image I.
- Encrypted image I_C obtained as:

$$I_C = BitXor\ [I_A, I_B] \quad (18)$$

- Send I_C and xG to Bob.

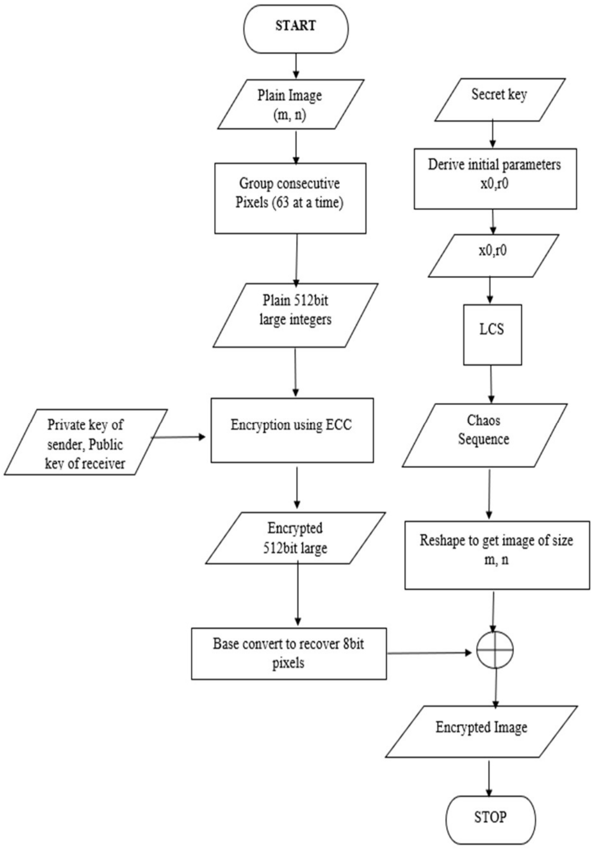

START
Plain Image (m, n)
Group consecutive Pixels (63 at a time)
Plain 512bit large integers
Private key of sender, Public key of receiver
Encryption using ECC
Encrypted 512bit large
Base convert to recover 8bit pixels
Secret key
Derive initial parameters x0,r0
x0,r0
LCS
Chaos Sequence
Reshape to get image of size m, n
Encrypted Image
STOP

Figure 2.1. Flow Chart Of Encryption Process.

3.4 Decryption

- At receiver end Bob uses the extracted key $K1$ to regenerate the pseudo random Chaos image $I_A{}'$. The original image is recovered from the received cipher image as follows:
- XOR the regenerated chaos image $I_A{}'$ with the received cipher image I_C to obtain intermediate image $I_B{}' = BitXor\ [I_A{}', I_C]$
- Using *FromDigits* Group pixels in I_B' to form big integers and pair to get encrypted points $P_c{}'$
- Calculate $K2' = n_B * xG$
- Now subtract $K2'$ from every encrypted point $P_c{}'$' to obtain Decrypted points $P_m{}'$
- $P_m' = P_c' - K2'$;
- Using *IntegerDigits*, get integer sequence ranging in 0-255 from P_m'
- Reshape to get decrypted image.

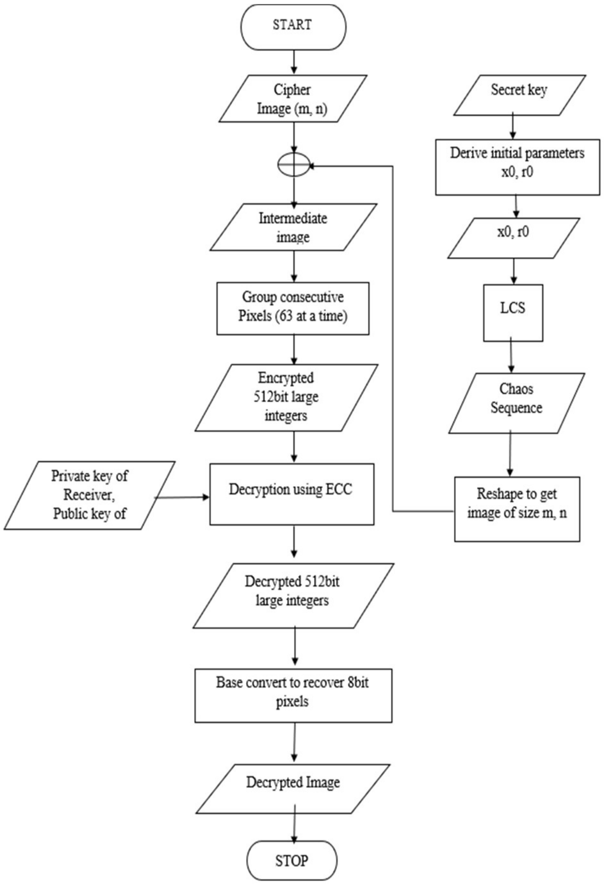

START
Cipher Image (m, n)
Intermediate image
Group consecutive Pixels (63 at a time)
Encrypted 512bit large integers
Private key of Receiver, Public key of
Decryption using ECC
Decrypted 512bit large integers
Base convert to recover 8bit pixels
Decrypted Image
STOP
Secret key
Derive initial parameters x0, r0
x0, r0
LCS
Chaos Sequence
Reshape to get image of size m, n

Figure 2.2. Flow Chart Of Decryption Process.

4. RESULTS AND DISCUSSIONS

In our experiment, we have used standard Brainpool 512-bit ECC curve $\mathrm{y}^2 \mathrm{mod}\ \mathrm{p} = (x^3 + \mathrm{ax} + \mathrm{b})\mathrm{mod}\ \mathrm{p}$ with following parameters:

p=89489622076502325516566028151591534221626096440983545113445971872000570104135524399179343041919569427654465303864273459379638943099239285360705346078 16947

a=62948605579730632276664213064763793240747157706227462271369104454503019142812760980279909684079839626911518536785638778342218340274397182380657258442 64138

b=32457890083289670592748495843420779165319090096375019183283236687361791765832634964635251284882826115598007735069737717977648114988349952343415308622 86627

$Generator(G_X, G_Y)$=(679205914042457517443564043126919508784315339010252188146802301273204748257985307754564744627286679

4936371522410774532686582484617946013928874296844351522,
6592244555240112873324748381429610341312712940326266331327445066687010545415256461097707483288650216992613090185042957716318301180159234788504307628509330)

Order of generator
n =8948962207650232551656602815159153422162609644098354511344597187200057010413418528378981730643524959857451398370029280583094215613882043973354392115544169

Private key of Bob
n_B=9619275968248211985332842594956369871234381391917297615810447731933374561248187549880587917558907265126128418967967816764706783230897486752408974005133

Public key of Bob P_B=
(559093065740956882543031860817394665823645932480056469674323622245113437121180431390259517423101920956842663682254230910744529800086849324159846843101049,
2687628544256193915322926560025669768994420507951673523285519876757954361251234973954559362562398273818907771202583044693743049889636577606972006551975671)

Private key of Alice
n_A=9426890448883247745626185743057242473809693764078951663494238777294707070023223798882976159207729119823605850588608460429412647567360897409117209856022401

Public Key of Alice P_A=
(7751118711104829465045639942070807370783729048087156698967198607925487952015814980426998029149611268753471042619483774234930071545732168049152355189964849,
5873502406727654222075919814064826101690644152314440390026929837164952123791022623994337003059298201835881846050997641303954599071246814465886192906194493)

The implementation was carried on i7,1.8GHz Dell laptop with 16GB RAM using Mathematica version 12 and MATLAB 2018. We took four greyscale images of size 256*256 available in standard *Mathworks* library for our experimentation. Figure 3 shows the original images and cipher images obtained after encryption respectively.

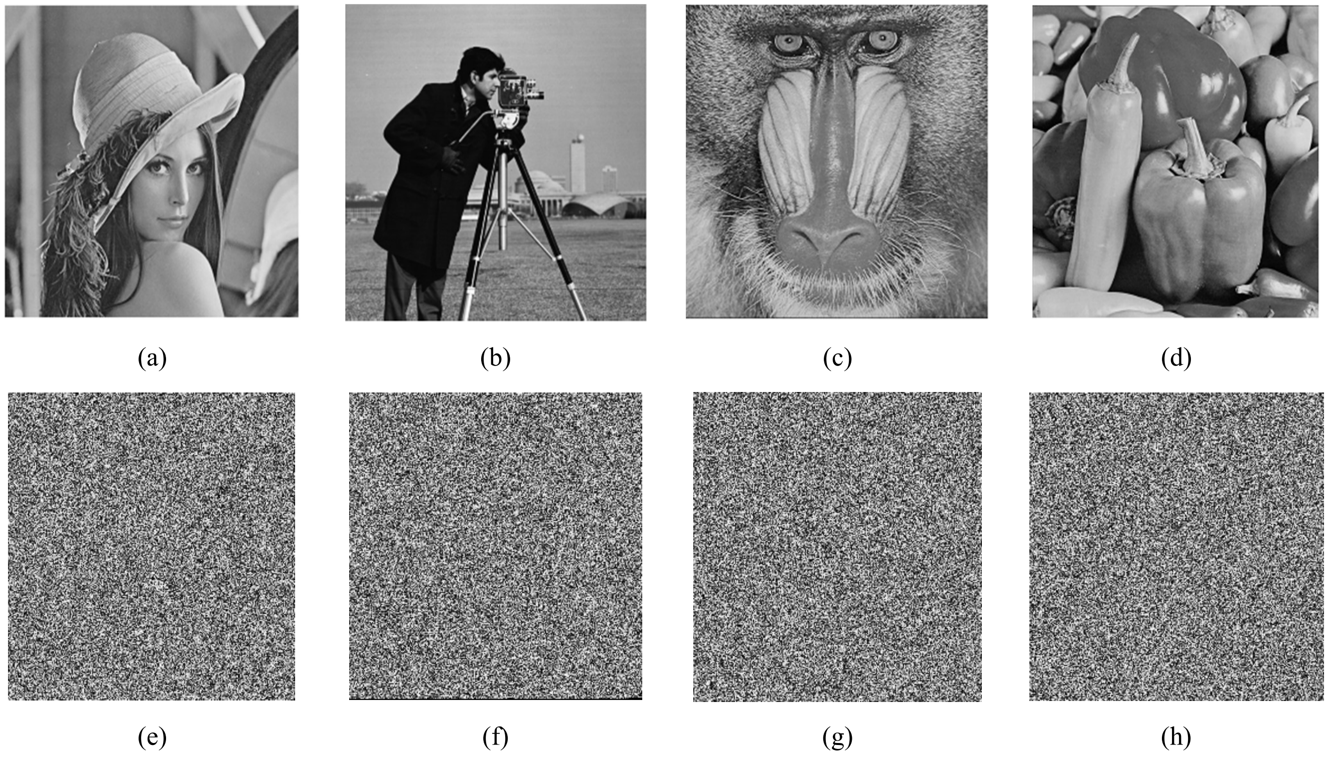


Figure 3. (A)-(D) Original Images Of Lenna, Cameraman, Baboon, Peppers; Є-(H) Cipher Images Of Lenna, Cameraman, Baboon And Peppers Obtained Using Proposed Method

4.1. Security Analysis

The efficiency of any encryption scheme is evaluated against some predefined security parameters. Security Analysis of an encrypted image is done by measuring its statistical properties such as Histogram, Inter-pixel Correlation, entropy analysis; Key Analysis such as Key space and key sensitivity Analysis; Diffusion Analysis such as NPCR, UACI, Avalanche effect, known-plaintext and Chosen-plaintext attack.

4.1.1 Brute-force attack and key space analysis

The security of any algorithm, inter-alia depends on key size. The Brute-force attack is carried out by trying every possible key combination on the cipher image until it is resolved into an intelligible image. On an average, it would require one-half of all possible key combinations. In our experiment we have used 512-bit key size for deriving both public and private keys. This results in key space of 2^{512} which is considered large enough to thwart any brute-force attack.

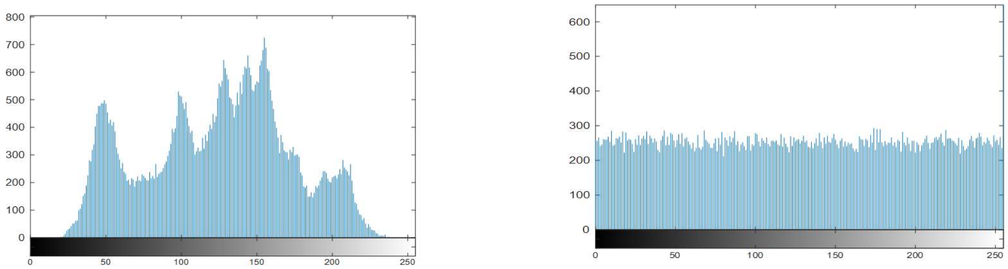
Given the hardness of Elliptic curve discrete logarithm problem together with the use of hybrid chaotic function for generating random sequence, a key size of 512-bits used in our proposed technique will efficiently throttle cryptanalysis, the key space analysis/ brute force attack.

Table 1. Brute-Force Analysis Of Proposed Technique.

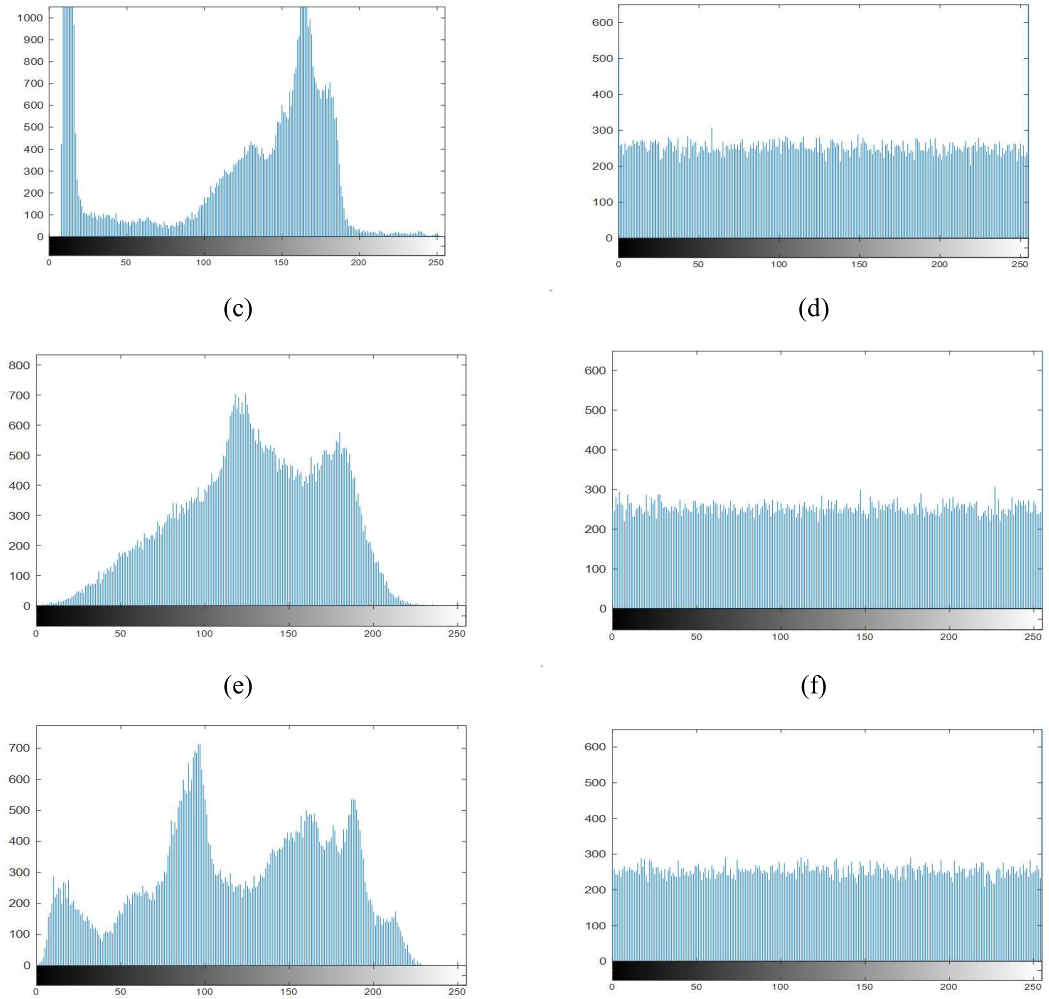
Key size	No. of Alternate keys	Time required for a successfull brute force attack by executing Million instructions/µs (Assuming)
2^{512}	2^{511}	6.37×10^{133} years

4.1.2 Histogram Analysis

Histogram tends to be popular tool for real-time image processing. It is represented by a discrete function h((r(k))=n(k), where r(k) is the kth intensity value and n(k) is the number of pixels in the image with intensity r(k). The histogram of an encrypted image should be uniform and have least correlation with the histogram of the original image. A good encryption algorithm should camouflage the characteristics of the original image. As it can be seen in Figure 4, the histogram of cipher images produced using our encryption technique doesn't show any similarity to histogram of original image. It indicates that all 256 intensities are equally probable and therefore the histogram of encrypted image is uniform.



(a) (b)



(g) (h)

Figure 4. Histogram Of Original Images, Cipher Images Of Lenna, Cameraman, Baboon, Peppers Respectively.

4.1.3 Time efficiency

The most time-consuming operation in proposed technique is Elliptic curve point multiplication. On sender side we use only two multiplication operations i.e., for key exchange, and for calculating K1 for encryption. The rest of the algorithm uses simple arithmetic operations like XOR; hence our technique is time efficient as compared to previously existing techniques.

4.1.4 Differential Attack Analysis

Diffusion is a measure of dependency of pixels of cipher image over plain image. A good encryption scheme should have strong diffusion such that even a slight change in plain image gives a totally different cipher image. Weak diffusion property can make a scheme vulnerable to differential attack and chosen-plain text attacks. To analyze diffusion property of we have used two parameters i.e., Number of pixels change rate (NPCR) Eq.19 and Unified average changing intensity (UACI) Eq. 20. The ideal values of NPCR and UACI are 99.61% and 33.46% respectively.

$$NPCR = (\sum_{i,j} D(i,j))/m * n) * 100 \quad (19)$$

$$UACI = \frac{1}{m*n} \frac{\sum_{i,j} C1(i,j) - C2(i,j)}{L} * 100 \quad (20)$$

Where $D(i,j)$ is 1 if pixel intensities $C1(i,j)$ and $C2(i,j)$ of two cipher images $C1$(actual cipher image), $C2$ (cipher image of plain image with 1 bit-modification) at position (i,j) are same else it's zero. L is total number of gray scale levels and $m \times n$ is size of image. A good encryption scheme renders NPCR, UACI values greater or equal to ideal values. Table 2 gives diffusion analysis of our algorithm.

Table 2. Differential Attack Analysis Using NPCR, UACI

Differential Attack Analysis				
	Image			
Parameter	Lenna	Baboon	Peppers	Cameraman
NPCR	99.60	99.60	99.61	99.58
UACI	33.70	33.63	33.858	33.59

4.1.5 Avalanche Effect Analysis

When a small change in plain text or encryption key results into completely different cipher text, then the encryption scheme is said to be resistant to differential attacks and has a good avalanche effect. Ideally the change should be greater than 50%. Avalanche effect can be measured by calculating mean square error (MSE) between the actual cipher image and cipher image obtained after modification of key or plain image. MSE is given by:

$$\frac{(\sum_{i,j} c1(i,j)-c2(i,j))^2}{m*n} \quad (21)$$

Where $c1(i,j)$ is pixel intensity at position (i,j) of actual cipher image $c1$, and $c2(i,j)$ is the pixel intensity of cipher image $c2$ with 1 bit modified key/plain image at position (i,j). Difference between two cipher images is visible if MSE is at least 33dB. Table 3 shows MSE analysis of cipher images obtained after 1 bit change in key, 1 bit change in plain image using our encryption scheme.

Table 3. Avalanche Effect Analysis

Avalanche Effect Analysis		
Image	**1 bit change in plain image**	
	MSE	**AE**
Lenna	39.169	50.18
Baboon	39.31	50.57
Peppers	39.431	50.18
Cameraman	39.004	50.25

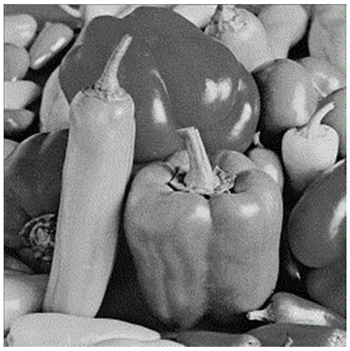
4.1.6 Key Sensitivity Analysis

An encryption algorithm should be highly sensitive to modifications in encryption and decryption keys. Even a slight change in encryption key should give a totally different cipher image. Similarly slight modifications in decryption key should make decryption of original image impossible. To analyze key sensitivity of our scheme we made slight changes in encryption and decryption keys, and initial parameters of our chaos system. The changed cipher images have been compared to cipher image obtained using actual keys (Figure 5). Number of pixels change rate (NPCR) Eqn.19 and Unified average changing intensity (UACI) Eqn. 20 are two analytical parameters to check the number of changes in modified cipher images. The ideal values of NPCR and UACI are 99.61% and 33.46% respectively.

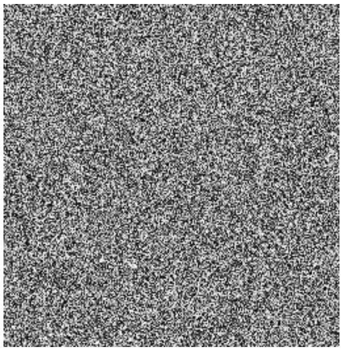
We performed key sensitivity analysis of our encryption scheme by using correct keys, keys with 1bit modification to obtain cipher images. Figure 5 and Table 4 shows the key sensitivity analysis performed on standard RGB test image of Baboon.

Table 4. Key Sensitivity Analysis

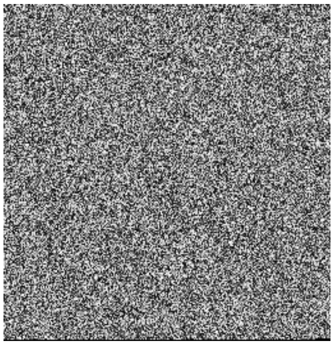
Key	Correct Key	Modified Key	Cipher image Using Modified Key	Difference between Actual and modified cipher image	NPCR	UACI
Hash value of Image	h	h+1	Fig. 5(c)	Fig. 5(d)	99.61	33.85
Receiver's Private key	n_B	n_B+1	Fig. 5(e)	Fig. 5(f)	99.61	33.85
Initial Parameter	x_0	x_0+0.1	Fig. 5(g)	Fig. 5(h)	99.55	33.73
Initial parameter	r	r+0.1	Fig. 5(i)	Fig. 5(j)	99.60	33.77
Shifting Factor of CTBCS	α	α+0.1	Fig. 5(k)	Fig. 5(l)	99.61	33.85
Sender's secret number	x	x+1	Fig. 5(m)	Fig. 5(n)	99.591	33.85



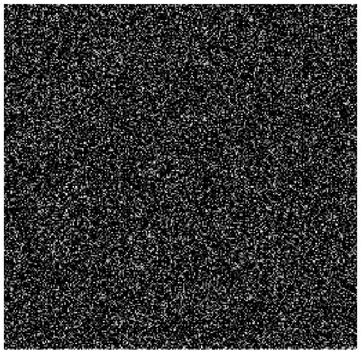
(a)



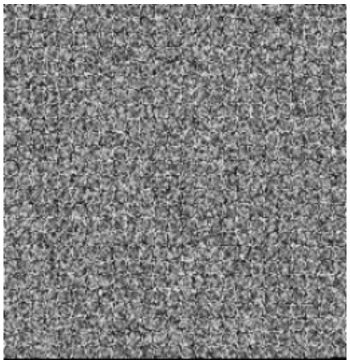
(b)



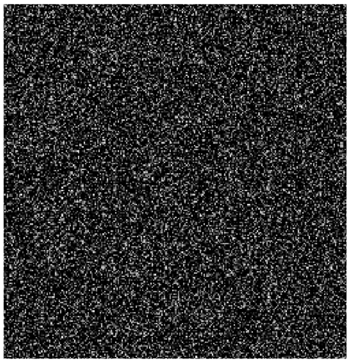
(c)



(d)



(e)



(f)

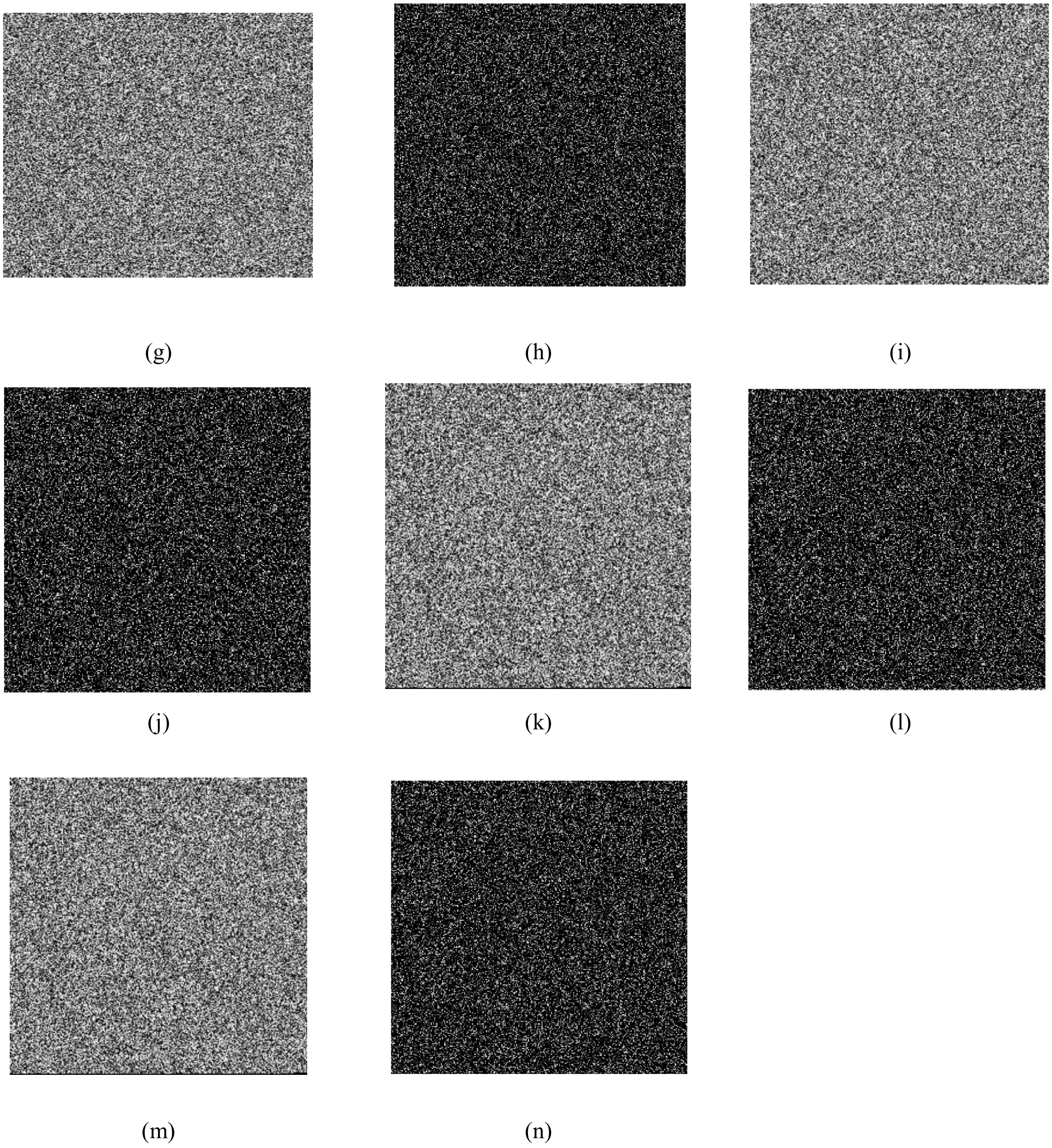


Figure 5. (A) Original Image; (B) Cipher Image, (C, E, G, I, K,M) Cipher Images With Modified Keys; (D, F, H, J, L,N) Are {B- (C, E, G, I, K)}

4.1.7 Entropy Analysis

Entropy of an image acts as a statistical measure of randomness of its pixel values. So greater the randomness in cipher image, better is the security. A good encryption technique will result in cipher images with entropy nearer to 8. Table 5 shows the entropy value of different test images before and after encryption using our technique.

Table 5. Entropy Analysis Of Various Images Before And After Encryption.

Image	Entropy	
	Original Image	Cipher Image
Lenna	7.4818	7.9975
Cameraman	7.0097	7.9968
Baboon	7.3897	7.9973
Peppers	7.6013	7.9972

4.1.8 Bit Analysis

Bit analysis is another widely used measure of randomness of data in encrypted image. It is done by calculating no. of occurrences of 1s and 0s in the binary equivalent of the encrypted data. A good cipher image has almost equal number of zeros and ones. Table 6 shows bit analysis of various cipher images obtained using proposed encryption technique.

Table 6. Bit Analysis Of Various Cipher Images Produced Using Proposed Technique.

Cipher Image	No of Zeros	No of Ones	Difference
Lenna	261927	262361	434
Cameraman	262011	262277	266
Baboon	262118	262170	52
Peppers	261636	262652	1016

4.1.9 Correlation Analysis

Pixel correlation serves as a measure of similarity between adjacent pixels. In general, neighbouring pixels in a plain image have similar intensity values hence the correlation between them is high. A high correlation means a correlation coefficient value tending to 1. A good encryption scheme must be able to reduce this correlation between neighbouring pixels considerably and give a correlation coefficient nearer to zero for encrypted image. Our technique is able to reduce the pixel correlation successfully. Table 7 shows horizontal, vertical, and diagonal pixel correlation of various test images before and after encryption.

A. Known, chosen plain-text attack

In general parlance, the known plain text attack assumes that the plain-text and corresponding cipher-text pair is known to a cryptanalyst or an adversary. In this case, he may build a repertoire of pairs of plane image and its corresponding cipher image in the form of block to deduce the key. In our proposed scheme, the security against this attack lies in randomness of sequence generated through hybrid chaos function to produce randomly distributed outputs and a large key space. The chaos sequence uses x_o as initial parameter and q_o as control parameter derived from SHA-512 , image dependent message digest value. Any change in the pixel value in the original image shall generate a different SHA-512 value and therefore alter the random sequence with new x_0 and q_0 values. This makes the relationship between the plane image and the cipher image, complex for any cryptanalyst or an adversary to deduce the encryption key. Therefore the proposed scheme will be resilient to the known plain-text attack. We have used two special images of all white and all black pixels, both of size 512*512 as chosen plaintext for cryptanalysis. Cipher images obtained are having uniform histogram distribution and entropies almost equal to ideal value of 8 as shown in Figure 6 and table 8. These highly random cipher images can't provide any relevance to our chosen all black and all white plain images. This proves that our encryption scheme is resilient to chosen plaintext attack as well.

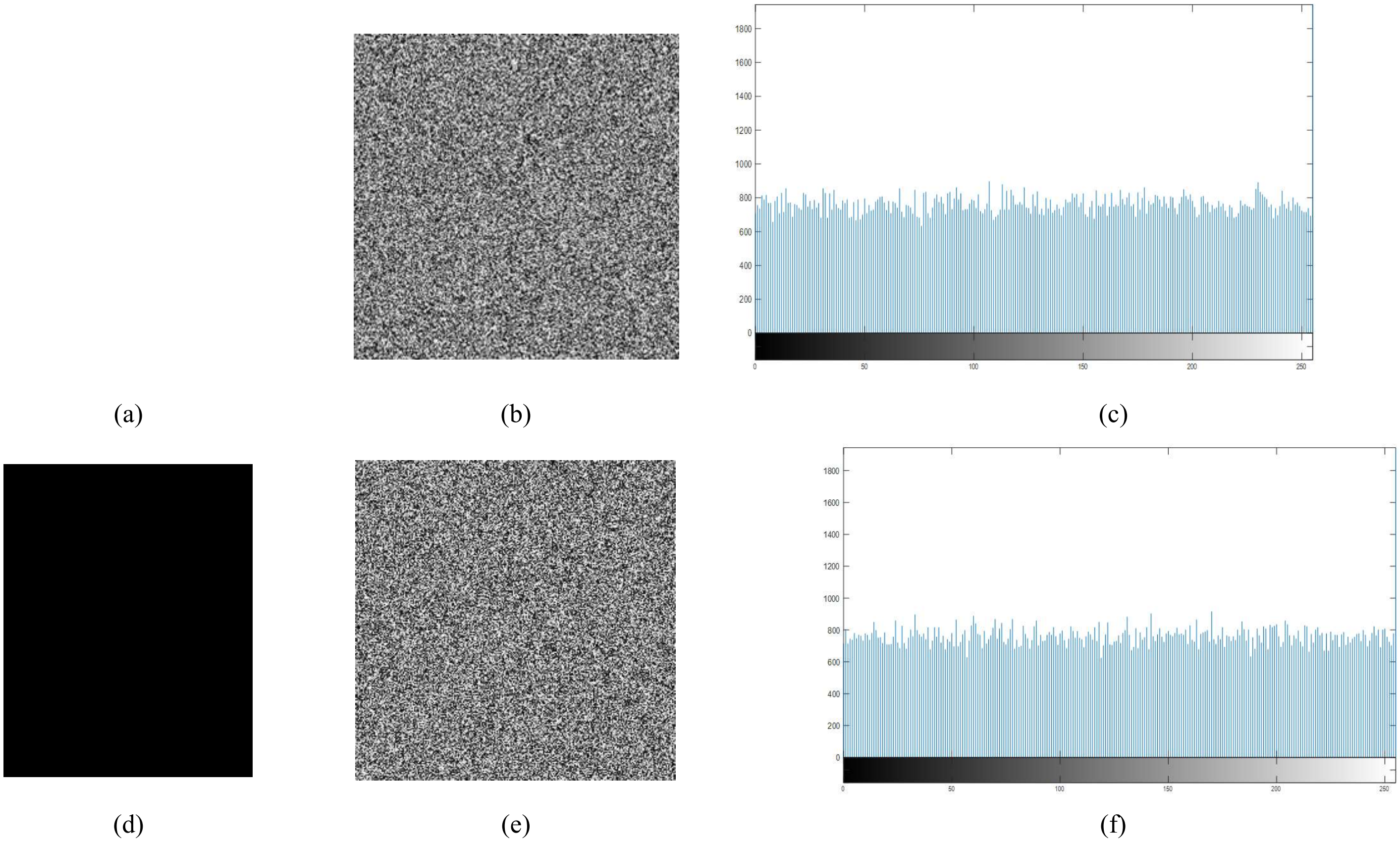
Table 7. Pixel Correlation Analysis Of Various Images Before And After Encryption

Correlation, Entropy Analysis of All Black and All White Images			
Cipher image of	Component	Value	Entropy
All Black	Horizontal	0.0030	7.987
	Vertical	- 0.00002	
	Diagonal	0.0019	

All White	Horizontal	0.0026	7.988
	Vertical	0.006	
	Diagonal	-0.005	

. Table 8. Correlation, Entropy Analysis Of All Black And All White Pixel Image

Image		**Correlation Coefficient**		
		Diagonal	Horizontal	Vertical
Lenna	Original image	0.8849	0.9106	0.9507
	Cipher image	-0.0003	0.0020	-0.0017
Cameraman	Original image	0.9087	0.9335	0.9592
	Cipher image	0.00043	-0.003	-0.00012
Baboon	Original image	0.6092	0.7169	0.6091
	Cipher image	-0.0015	0.0034	0.0015
Peppers	Original image	0.9141	0.9460	0.9538
	Cipher image	0.0025	-0.005	0.0018



(a) (b) (c)

(d) (e) (f)

Figure 6. (A) All White Image; (B)Cipher Image Of (A); (C) Histogram Of (B); (D) All Black Image; (E) Cipher Image Of (D); (F) Histogram Of (E).

4.2 Comparison with Other ECC based schemes

In order to evaluate performance of proposed technique, a comparison with some of the recently proposed popular ECC based image encryption schemes has been done. Analysis shows that our proposed scheme uses only 2 ECC multiplications, taking minimum time of 0.11 seconds. Proposed technique is time efficient while giving results at par with other techniques. Table 9 gives a comparison of different techniques based on encryption of a 256 x 256-pixel image.

5. CONCLUSION AND FUTURE SCOPE

In this paper, we proposed a simple image encryption technique using hybrid logistic-cosine-sine (LCS) chaos map and elliptic curve cryptography. The use of LCS cosine transform based chaos system CTBCS offers a complex mapping to generate randomly distributed outputs and a large key space, which helps to thwart known plain text attack, prevalent in some of the common image encryption schemes. The chaos system generates its control parameters from the key securely exchanged between sender and receiver. Encryption is done by XORing a random chaos sequence with ECC encrypted plain image. To avoid key space analysis attack or brute force attacks, a key size of 512 bits is recommended. Instead of working on individual image pixels, a group of pixels is encrypted together where the no of pixels to be grouped is governed by key length. Experimental results show that our scheme is cost and time efficient as entire image encryption consumes only 0.3 sec making it comparable with the fastest known ECC and chaos-based schemes. This makes the proposed scheme feasible for use in real-time scenarios. Security Analysis shows that proposed scheme is resilient against Brute-force, known plain-text and statistical analysis attacks. In future, the proposed encryption algorithm can be optimised further to improve time efficiency and randomness/ entropy of encrypted image. Use of other chaos mapping functions such as Henon, tent map, Arnold map, Lorenz map will be explored in the future to improve ergodicity. The proposed technique may be extended to encrypt Colour, biomedical images.

Table 9. Comparison With Other Techniques

Parameters	Ref [25]	Ref [20]	Ref [2]	Ref [3]	Proposed
Encryption Time (in secs.)	0.29	-	0.0518	0.395	0.3
No. of ECC point multiplication, time taken per multiplication in sec.	-	2, 18.3	2048,0.077	-	2, 0.11
Horizontal Correlation	0.0023	-0.0013	-0.0017	0.0010	0.0020
Vertical Correlation	0.047	0.0025	-0.0148	0.0017	-0.0017
Diagonal Correlation	0.0206	0.0014	0.0073	0.0125	-0.00033
Entropy	7.99884	7.9990	7.9974	7.9973	7.9975

REFERENCES

[1] Matthews RAJ (1989) On the derivation of a 'chaotic' encryption algorithm. Cryptologia 13(1):29–42.

[2] Laiphrakpam, Dolendro Singh, and Manglem Singh Khumanthem. "A robust image encryption scheme based on chaotic system and elliptic curve over finite field." Multimedia Tools and Applications 77.7 (2018): 8629-8652.

[3] Abd El-Latif, A. A., & Niu, X. (2013). A hybrid chaotic system and cyclic elliptic curve for image encryption. AEU-International Journal of Electronics and Communications, 67(2), 136-143.

[4] Francois, M., Grosges, T., Barchiesi, D., & Erra, R. (2012). A new image encryption scheme based on a chaotic function. *Signal Processing: Image Communication*, *27*(3), 249-259.

[5] Lan, Rushi, et al. "Integrated chaotic systems for image encryption." Signal Processing 147 (2018): 133-145.

[6] M. Ahmad and M. Shamsher Alam, A New Algorithm of Encryption and Decryption of Images Using Chaotic Mapping, International Journal on Computer Science and Engineering, Vol.2 (1), pp. 46-50, 2009.

[7] Hanchinamani, G., & Kulkarni, L. (2015). An efficient image encryption scheme based on a Peter De Jong chaotic map and a RC4 stream cipher. 3D Research, 6(3), 30.

[8] Bansal, R., Gupta, S., & Sharma, G. (2017). An innovative image encryption scheme based on chaotic map and Vigenère scheme. Multimedia Tools and Applications, 76(15), 16529-16562.

[9] Zeghid, M., Machhout, M., Khriji, L., Baganne, A., & Tourki, R. (2007). A modified AES based algorithm for image encryption. International Journal of Computer Science and Engineering, 1(1), 70-75.

[10] Sun, Q., Chang, S. F., Maeno, K., & Suto, M. (2002, May). A new semi-fragile image authentication framework combining ECC and PKI infrastructures. In 2002 IEEE International Symposium on Circuits and Systems. Proceedings (Cat. No. 02CH37353) (Vol. 2, pp. II-II). IEEE.

[11] Gong, Lihua, et al. "An optical image compression and encryption scheme based on compressive sensing and RSA algorithm." Optics and Lasers in Engineering 121 (2019): 169-180.

[12] ur Rehman, Aqeel, et al. "A color image encryption technique using exclusive-OR with DNA complementary rules based on chaos theory and SHA-2." Optik 159 (2018): 348-367.

[13] Miller, V. S. (1985, August). Use of elliptic curves in cryptography. In *Conference on the theory and application of cryptographic techniques* (pp. 417-426). Springer, Berlin, Heidelberg.

[14] Soleymani, Ali, et al. "A binary grouping approach for image encryption based on elliptic curves over prime group field." 2013 IEEE 11th Malaysia International Conference on Communications (MICC). IEEE, 2013

[15] Singh, Laiphrakpam Dolendro, and Khumanthem Manglem Singh. "Image encryption using elliptic curve cryptography." Procedia Computer Science 54 (2015): 472-481.

[16] Gupta, Kamlesh, et al. "An ethical way of image encryption using ECC." 2009 First International Conference on Computational Intelligence, Communication Systems and Networks. IEEE, 2009.

[17] Kamrani, A., Zenkouar, K., & Najah, S. (2020). A new set of image encryption algorithms based on discrete orthogonal moments and Chaos theory. *Multimedia Tools and Applications*, 1-17.

[18] Sasikaladevi, N., Geetha, K., Sriharshini, K., & Aruna, M. D. (2019). RADIANT-hybrid multilayered chaotic image encryption system for color images. *Multimedia Tools and Applications*, *78*(9), 11675-11700.

[19] Farah, M. B., Guesmi, R., Kachouri, A., & Samet, M. (2020). A novel chaos based optical image encryption using fractional Fourier transform and DNA sequence operation. *Optics & Laser Technology*, *121*, 105777.

[20] Zhang, X., & Wang, X. (2018). Digital image encryption algorithm based on elliptic curve public cryptosystem. *IEEE Access*, *6*, 70025-70034.

[21] Dawahdeh, Z. E., Yaakob, S. N., & bin Othman, R. R. (2018). A new image encryption technique combining Elliptic Curve Cryptosystem with Hill Cipher. *Journal of King Saud University-Computer and Information Sciences*, *30*(3), 349-355.

[22] Toughi, S., Fathi, M. H., & Sekhavat, Y. A. (2017). An image encryption scheme based on elliptic curve pseudo random and advanced encryption system. *Signal processing*, *141*, 217-227.

[23] Luo, Y., Ouyang, X., Liu, J., & Cao, L. (2019). An image encryption method based on elliptic curve elgamal encryption and chaotic systems. *IEEE Access*, *7*, 38507-38522.

[24] Gupta, K., & Silakari, S. (2011). Efficient hybrid image cryptosystem using ECC and chaotic map. *Int. J. Comput. Appl*, *29*(3), 1-13.

[25] Singh, L. D., & Singh, K. M. (2015). Image encryption using elliptic curve cryptography. *Procedia Computer Science*, *54*, 472-481.

[26] Bakhtiari, S., Ibrahim, S., Salleh, M., & Bakhtiari, M. (2014, August). JPEG mage encryption with Elliptic Curve Cryptography. In *2014 International Symposium on Biometrics and Security Technologies (ISBAST)* (pp. 144-149). IEEE.

[27] Soleymani, A., Nordin, M. J., & Ali, Z. M. (2013). A novel public key image encryption based on elliptic curves over prime group field. *Journal of Image and Graphics*, *1*(1), 43-49.

[28] Yadav, V. K., Malviya, A. K., Gupta, D. L., Singh, S., & Chandra, G. (2012). Public key cryptosystem technique elliptic curve cryptography with generator g for image encryption. Int. J. Comput. Technol. Appl, 3(1), 298-302.

[29] Gupta, K., Silakari, S., Gupta, R., & Khan, S. A. (2009, July). An ethical way of image encryption using ECC. In *2009 First International Conference on Computational Intelligence, Communication Systems and Networks* (pp. 342-345). IEEE.

[30] Lan, Rushi, et al. "Integrated chaotic systems for image encryption." Signal Processing 147 (2018): 133-1.