Hybrid Image Encryption Using Elliptic Curve Cryptography, Hadamard Transform and Hill Cipher

Desam Vamsi*
School of Computer Science and Engineering, VIT-AP University, Andhra Pradesh, India.
E-mail: d.vamsi1@gmail.com

Pradeep Reddy CH
School of Computer Science and Engineering, VIT-AP University, Andhra Pradesh, India.
E-mail: pradeep.ch@vitap.ac.in

Received September 11, 2021; Accepted December 10, 2021
ISSN: 1735-188X
DOI: 10.14704/WEB/V19I1/WEB19160

Abstract

In this digital world, communication systems have witnessed abundant usage of media over the platforms. Among these, providing security in transmission of images is highly important, and attained a lot of research interest due to its high consideration in both the industry and the academic community. This paper proposes a hybrid asymmetric image encryption algorithm using Elliptic curve cryptosystem (ECC), Hadamard transform and Hill cipher algorithms. Based on the Diffie–Hellman public key exchange method a point on the elliptic curve is selected and agreed between both the sender and receiver. The key relies upon the ECC and it is difficult to resolve the ECDLP to get it. The proposed algorithm involves two stages of encryption, primarily, the XOR function is applied on the Elliptic curve Diffie-Hellman (ECDH) shared secret key and the hadamard image. In the subsequent stage, ECC is combined with the hill cipher algorithm. Encryption and decryption uses self-invertible key matrix, hence the process of finding inverse key becomes redundant during decryption which improves the speed of execution. It also enhances the security and efficiency compared to original hill cipher method. The results are compared with other ECC methods proves that the current cryptosystem attains large key space, highly key sensitive, low correlation and can resist against differential and statistical attacks.

Keywords

Asymmetric Image Encryption, Cryptosystem, Diffie–Hellman Public Key Exchange, Elliptic Curve Diffie-Hellman (ECDH), Elliptic Curve Discrete Logarithm Problem (ECDLP).

Introduction

Cryptography is the mathematical method used to secure images and text from adversaries and improve the security in communication mediums. Sender performs the encryption where the plain image is converted into an encrypted image before it is sent to the receiver via internet. At the receiver's end, decryption is performed where the encrypted (ciphered) image returns to its plain image. As utilization of images in the web has increased tremendously, its protection of content has become an important topic. Hence, the researchers introduced various types of encryption methods for images (Luo et al., 2018; Song & Lee, 2021; Kaur et al., 2020; Jiang et al., 2021; Wang et al., 2019). Two forms of encryption methods are used in these algorithms; symmetric (private-key) and asymmetric (public-key) encryption (Simmons, 1979). In symmetric encryption, the same key is used for the process of encryption and decryption. These algorithms are efficient and fast in execution, particularly for large quantities of information such as images (Huang et al., 2019; Luo et al., 2018; Setyaningsih et al., 2020). But, key distribution and management is a major drawback in symmetric encryption. The key must be reliably transmitted over the network, but it can be intercepted during transmission by attackers. In fact, as the users count increases there will be a sudden increase in number of keys, which is a burden to the network.

The asymmetric (Public Key Encryption-PKE) encryption subdues these issues where two keys (public, private) are used separately for encryption and decryption. The private key is hard to obtain from the public key. So, in this encryption there is no need of exchanging the private key as the receiver has his own private key. Hence, the issue with the distribution of key does not prevail in the encryption. It can also provide a digital signature feature that cannot be attained with a symmetric encryption. The services that are offered by digital signatures are non-repudiation, message integrity and authentication. In public key encryption, the most difficult mathematical problems are; discrete logarithmic problem and factorization problem. Such two problems are used in Digital signature algorithm (DSA) and Rivest–Shamir–Adleman (RSA).

A modern PKE is developed by Miller (Miller, 1986) and Koblitz (Kobliz, 1987) in 1985 named as elliptic curve cryptography (ECC), which increases the efficiency of many techniques. In addition, cryptographers have observed that they are able to acquire computational efficiency in execution with very less key-size and provide high security using ECC as compared to other algorithms. The ECC has become more interesting, as it solves discrete logarithm problem by correctly selecting the elliptic curve that has no sub exponential technique. So, ECC provides same security level even it uses small parameters

when compared with other algorithms. This makes Elliptic curve cryptography more efficient with faster computations, and less storage space, processing power and bandwidth. In various applications, the Elliptic curve Diffie-Hellman algorithm (Diffie & Hellman, 1976) is commonly used for key exchange scheme. For, symmetric key encryption algorithms like AES, DES and RC4 the shared secret keys can be exchanged by using this algorithm. Several ECC based image encryption techniques are proposed. In this paper (Shankar & Eswaran, 2016) the author presented a novel ECC based image encryption using genetic method to achieve best optimized key. In this approach, the image is encrypted pixel by pixel using ECC. However, the usage of ECC on every pixel and finding the optimal key are highly expensive in computations. In (Luo et al., 2019), a new ECC-Elgamal encryption system is designed to enhance the security in existing system of permutation-diffusion.

This algorithm not only hold against chosen-plaintext and known-plaintext attacks, but also increases pixel distribution randomness. However, it is a complex architecture and time consuming. In (Khoirom et al., 2018) the author gave cryptanalysis simulation results of the elliptic curve encryption technique proposed in (Tawalbeh et al., 2013) and stated that the attack using Pollard's Rho or Baby-step giant-step (BSGS), naive method would be practically impossible if the suggested Elliptic curve (EC) parameters provided by the organizations like Brainpool or National Institute of Standards and Technology (NIST).

Hill cipher algorithm (Hill, 1929) invented in 1929 by L.S. Hill is one of the known secret key encryption that has a simple structure, fast execution speed, and high throughput. It has poor as the sender and receiver share the same private key while transmitting the data over unsecured channel. Several researchers made an effort to develop and enhance the security of hill cipher technique. In (Ismail et al., 2006) the author designed an approach called Hill Multiplying Rows by Initial Vector (HillMRIV) algorithm. This algorithm generates different keys for encrypting each block of plaintext rather than using single matrix key for all plaintext blocks. This improves hill cipher algorithm security, but fails when the plaintext blocks contains only zeros. A new approach Advanced hill algorithm (AdvHill) (Acharya et al., 2007) is designed, to solve the problem of decryption when the key matrix inverse doesn't exist. AdvHill algorithm generates an involutory key matrix, and the encryption and decryption are performed with the same key matrix. So, it decreases the computations because there is no need to obtain the inverse key matrix by the recipient and the improved cipher randomization increases the algorithm performance compared to the initial Hill cipher. In (Khazaei & Ahmadi 2017) the author strengthen the divide-and-conquer ciphertext only attack on hill cipher using Chinese Remainder Theorem on the code complexity of O(d13d) with a slightly higher data complexity cost. Evaluating the results of the proposed system or justifying their optimality on the basis of reasonable assumptions

about computational complexity is still an open issue. In (M Essaid et al., 2012) designed an algorithm to make the encryption technique more efficient, by encrypting all kinds of images pixel by pixel, even with the images having black background or the adjacent pixels of image having high correlation.

Hadamard transform is one of the orthogonal transforms of matrix which is specially used because of having only two entries i.e., +1 and -1. Therefore this transformation has very fast computational speed and requires very low memory. In total, the efficiency is better than any other transform methods. Several works (Harwit, 2012; Qu et al., 2021; Zheng & Huang, 2018; Prajwalasimha, 2019), have studied on the Hadamard transform and found few applications that are commonly employed in transmission, digital compression, signal processing, data encryption and communication technology. In (Devi & Singh, 2020) the author proposed a novel copyright protection technique named as Red-Cyan Anaglyph image watermarking using Discrete Wavelet, Fast Walsh Hadamard, Arnold Transform and Singular Value Decomposition. By using this technique the hidden data is being kept secret will not be lost easily. The author proposed a new approach (Wang et al., 2019) using discrete Walsh–Hadamard transform for segmentation and fast detection of partial blur regions in actual images. This approach is simple to execute and provide high efficiency due to its quick sequence transforms.

In this paper a new encryption algorithm is proposed using Elliptic Curve Cryptosystem (ECC), Hill Cipher (HC) and Hadamard Transform (HT). The proposed algorithm provides hybrid encryption with two levels of encryption, one with Elliptic curve key and Hadamard transformation on image and second with Elliptic curve hill cipher. A key is generated from Elliptic curve and a Hadamard transformed image is encrypted with XOR, and followed by Elliptic curve hill cipher encryption. The encrypted image is transmitted to the receiver. At the receiver end, the decryption is performed by self-invertible hill cipher and that is XOR with the elliptic curve key to obtain the hadamard image. Further, the inverse hadamard is applied to retrieve the original image. This algorithm has the advantage of providing more security with hybrid encryption which is hard to crack by the intruders.

This paper covers the following sections. Section 2 give introduction to elliptic curve function. The original Hill cipher algorithm is explained in section 3. Section 4 describes the detailed view of Hadamard transform. The proposed hybrid encryption algorithm is presented in section 5. Section 6 contains security analysis for some measures. Next, an algorithm for proposed encryption is presented in section 7. Finally, section 8 ends with the conclusion and the advantages of proposed algorithm.

Background

Elliptic Curve Operations

a. Structure

The representation of Elliptic curve is $E_p(a, b)$ where a, b are confined to $mod\ p$ and p is a prime number. The Weierstrass normal form, the fundamental Elliptic curve E utilized for cryptography is in the form E: $y^2 = (x^3 + ax + b)\ \ mod\ p$ over a prime field F_p which is represented in Fig. 1. Here $a, b \in F_p$, $p \neq 2, 3$ and the curve should satisfies the condition $4a^3 + 27b^2 \neq 0$ called Non-Singular Elliptic Curve, then it has 3 distinct roots which is suitable for ECC. The elliptic curve group $E_p(a, b)$ includes all (x, y) points which satisfies the elliptic curve E along with an additional point O called as point to infinity.

b. Operations

Elliptic curve arithmetic operations performed on single point or two distinct points on the curve as parameters and generate a new point that lies on the curve. These operations made ECC powerful and an effective way in cryptography.

Point Addition: Adding two points on the curve results a new point which lies on the curve is calculated as follows. For suppose, from the Figure. 1 there are two points P and S which lies on the curve, adding P and S results the point R. When addition is performed on P and S, draw a straight line joining P and S and extend the line to some other point which touches the curve called R. In general the point addition of Elliptic curve is defined as:

$$P\ +\ S\ =\ R \tag{1}$$

Suppose, $P(x_p, y_p)$ and $S(x_s, y_s)$ and gives $R(x_r, y_r)$.

The R coordinates are functioned as:

$$x_r = \lambda^2 - x_p - x_s \tag{2}$$

$$y_r = \lambda(x_p - x_r) - y_p \tag{3}$$

Where,

$$\lambda = \frac{(y_s - y_p)}{(x_s - x_p)},\ \ if\ \ P \neq S \tag{4}$$

Point Doubling: Point doubling is nothing but point addition, if P and S are pointing the same point on the curve then the slope of the curve λ is calculated as given in this section. In general the point doubling of Elliptic curve λ function is defined as:

$$\lambda = \frac{(3x_p^2+a)}{2y_p}\ ,\ if\ \ P = S \qquad (5)$$

Point Multiplication: Only scalar multiplication is possible on the points. It is same as point addition, perhaps point doubling, for n times it is applied, and calculate nP where the scalar positive value is n.

$$nP = \sum_{i=1}^{n} P = R \qquad (6)$$

The computations in the elliptic curves applies itself naturally to the modular arithmetic operation. This guarantees the curve gets characterized over a field that makes ECC calculation attainable. The mathematical operations directly require modulo p, to shift the curves from real numbers to finite fields, where p be a prime value, to uniformly circulate values over the finite modular field.

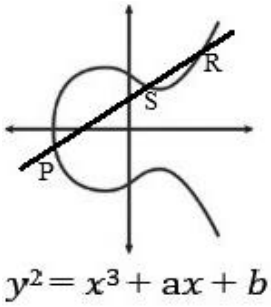

Figure 1 Points on Elliptic curve

Hill Cipher

In 1929 Hill cipher was innovated by Lester Hill as a symmetric block cipher method. In this method, the ciphering and deciphering should be performed with the same key matrix that is shared by both sender and receiver. The principle used in this method is relied on assigning each alphabet with a number, for instance, a=0, b=1, c=2, d=3 … z=25. According to the size of key matrix $n \times n$ the plaintext is divided into blocks of size n. For example, if the size of key matrix (Q) is 2×2 then the plaintext block (T_P) size should be 2×1, and then the encryption is performed with both the matrices Q and T_P the resultant matrix is obtained known as ciphertext block (T_C) of size 2×1 with numeric values [17]. The

receiver should calculate the inverse of the key matrix (Q) i.e. Q^{-1} where, $Q^{-1} * Q = I$ (Identity matrix), to decipher the ciphertext (T_C) message.

Mathematical notation:

To encrypt the plaintext

$$T_C = Q \times T_P \ (mod\ 26) \tag{7}$$

To decrypt the ciphertext

$$T_P = Q^{-1} \times T_C \ (mod\ 26) \tag{8}$$

Hadamard Transform

Hadamard Transform (HT) is also termed as Walsh- Hadamard Transform (WHT) which is a generalized series of Fourier transforms. HT considers matrices of unitary type i.e. +1 or -1. Hadamard unitary matrix of m order is the $M \times M$ matrix, where $M = 2^m$, which is formed using the following iteration rule.

$$H_m = H_1 \otimes H_{m-1} \tag{9}$$

Where,

$$H_1 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{10}$$

And $\otimes$ represents the Kronecker product of the given two matrices

$$C \otimes D = \begin{bmatrix} C(1,1)D & C(1,2)D & \cdots & C(1,M)D \\ \vdots & \vdots & \vdots & \vdots \\ C(M,1)D & C(M,2)D & \cdots & C(M,M)D \end{bmatrix} \tag{11}$$

Where,

$P(i,j)$ is the (i,j) element of P and $i,j = 1,2,\ldots M$.

According to E.q (9) – (10)

$$H_2 = H_1 \otimes H_1 = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad (12)$$

and,

$$H_3 = H_1 \otimes H_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} H_2 & H_2 \\ H_2 & -H_2 \end{bmatrix} \text{ for, m} = 3 \quad (13)$$

The orthogonality of H_m is not hard to demonstrate where, $m = 1, 2$ ...

$$H_m^{-1} = H_m^T = H_m \quad (14)$$

A vector u of M-sample matrix and $M = 2^m$ the transform pair (u, v) is

$$v = H_m\, u, \;\; u = H_m\, v \quad (15)$$

The Hadamard transformation in two dimensions is carried out by

$$V = H_m\, U\, H_m, \;\; U = H_m\, V\, H_m \quad (16)$$

The results obtained using hadamard transform takes less computational complexity for calculating the transform coefficients. It can be stated as one of the fastest algorithm due to its time complexity O($M \log_2 M$) (Anil K, 1989).

Proposed Method

In this section the proposed encryption and decryption algorithm is discussed in detail. The proposed algorithm uses ECC, Hadamard transform and Hill cipher methods. The encryption and decryption of the image using this algorithm is described as follows:

Encryption: First at the sender side, the Elliptic curve is used to generate the key. By using the key, XOR operation is performed with Hadamard transformed image. Later, by using ECC key a self-invertible matrix is generated and perform hill cipher encryption and send the encrypted image to the receiver's end.

Decryption: At the receiver's end encrypted image is decrypted by applying hill cipher decryption. In this algorithm there is no requirement to compute the inverse key matrix because the generated key is self-invertible. Later, the XOR operation is performed with

the initial EC key and the inverse hadamard is applied to the resultant image then, the original image is obtained.

- The hardness of the EC Discrete Logarithm Problem (EC-DLP) provides the security to EC based algorithms. It is defines as follows: Let $E_p\ (a, b)$ represents elliptic curve, S is a point on EC so that $S = n.G$ Where, G is base point and n is random integer.
- It is difficult to obtain the discrete logarithm of S with a publicly known G (Li et al., 2012). The following EC Diffie Hellman key exchange protocol is used to exchange secret key between the parties (sender & receiver).

Step 1: Assume that two users are in communication, the User X (sender) transmit the information in the form of an image (P_I) to the User Y (receiver) through an insecure medium. Initially the elliptic curve E should be agreed by both the Users, and the domain parameters over F_p are shared $\{p, a, b, G, n, h\}$, where p is large prime integer, $a, b\ \in F_P$ are the coefficients specifying an elliptic curve E, G is the base point on E, n is a random integer specifying F_P, h is the cofactor. Later, from the interval $[1,2,3 \dots, p-1]$ each user has to select their private key randomly; z_X for User X (sender) and z_Y for User Y (receiver).

The public key generation of User X and Y are as follows:

$$Pu_X = z_X . G \quad (17)$$

$$Pu_Y = z_Y . G \quad (18)$$

In order to obtain the initial key $Q_I = (\alpha, \beta)$, every user performs multiplication on their private key with the other user's public key.

$$Q_I = z_X . Pu_Y = z_Y . Pu_X = z_X . z_Y . G = (\alpha, \beta) \quad (19)$$

With public keys Pu_X or Pu_Y it is difficult to find Q_I, as it involves solving of ECDLP for obtaining the z_X or z_Y private keys which is unattainable. Then Q_1 and Q_2 can be computed as:

$$Q_1 = \alpha . G = (q_{11}, q_{12}) \quad (20)$$

$$Q_2 = \beta . G = (q_{21}, q_{22}) \quad (21)$$

By performing this Elliptic curve key generation, both the users generate the keys Q_1 and Q_2 and it is written in the form of 2×2 key matrix Q_m.

$$Q_m = \begin{bmatrix} Q_1 \\ \\ Q_2 \end{bmatrix} \tag{22}$$

Step 2: Hadamard transforms that are discussed in section 4, are applied on an original image (P_I) with 256 × 256 pixels. The resultant hadamard transform image is obtained $H(P_I)$.

Step 3: In the next step, XOR operation is performed on the key Q_m and $H(P_I)$.

$$R_I = Q_m \oplus H(P_I) \tag{23}$$

Step 4: Further, the sender and receiver should generate another secret key Q_n for encryption and decryption by using the following procedure. The key matrix inverse does not always exist and also if the inverse of the key matrix is not possible, then the ciphertext cannot be decrypted by the recipient. The self-invertible matrix is introduced in order to address this issue, and the need to calculate the inverse key matrix doesn't arise, for encryption and decryption the same key is used (the key matrix Q is self-invertible, if $Q = Q^{-1}$) in the communication. This algorithm is implementing on 256×256 pixel images (gray scale). The image (R_I) pixels are divided into 4-blocks for encryption where, ($R_I = (P_{I1}; P_{I2}; P_{I3}; \ldots)$). Then, each user generates a self-invertible key matrix Q_n of size 4×4 discussed in [19].

Suppose, $Q_n = \left[\begin{array}{cc|cc} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ \hline q_{31} & q_{32} & q_{33} & q_{34} \\ q_{41} & q_{42} & q_{43} & q_{44} \end{array}\right]$ is a self-invertible matrix subdivided as $Q_n = \left[\begin{array}{c|c} Q_{11} & Q_{12} \\ \hline Q_{21} & Q_{22} \end{array}\right]$

Let us, assume $Q_{11} = \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix}$ then, by solving the following conditions the remaining sub-division values of the secret key matrix Q_n is generated.

$$Q_{12} = I - Q_{11} \tag{24}$$
$$Q_{21} = I + Q_{11}, \text{ and} \tag{25}$$
$$Q_{11} + Q_{22} = 0. \tag{26}$$

Where, I= identity matrix

Next, divide pixel image values into four size blocks, each block is transformed into a column vector of 4×1 size, respectively $(P_{I1};\ P_{I2};\ P_{I3};\ \ldots)$, later multiply each vector of image with a self-invertible key matrix Q_n and perform modulo 256 to obtain the ciphered vectors $(C_{I1};\ C_{I2};\ C_{I3};\ \ldots)$. By using the values in the ciphered vectors, construct a ciphered image C_I and transmit it to the receiver. For each block the following computations are repeated:

Let $P_{I1} = \begin{bmatrix} p_{11} \\ p_{21} \\ p_{31} \\ p_{41} \end{bmatrix}$ then,

$$C_{I1} = Q_n \times P_{I1} = \begin{bmatrix} c_{11} \\ c_{21} \\ c_{31} \\ c_{41} \end{bmatrix} \tag{27}$$

When the receiver receives the ciphered image C_I, the decryption process begins by dividing the image pixel values into blocks of each size four, and then each block of C_I is arranged into 4-rows column vector. Later, multiply each vector of ciphered image $(C_{I1};\ C_{I2};\ C_{I3};\ \ldots)$ with a self-invertible key matrix Q_n and perform modulo 256 to obtain the plain image vectors $(P_{I1};\ P_{I2};\ P_{I3};\ \ldots)$. By using these $(P_{I1};\ P_{I2};\ P_{I3};\ \ldots)$ values the image R_I is obtained.

$$P_{I1} = Q_n \times C_{I1} = \begin{bmatrix} p_{11} \\ p_{21} \\ p_{31} \\ p_{41} \end{bmatrix} = R_I \tag{28}$$

Step 5: The XOR operation is performed with the key Q_m and R_I to obtain the result, hadamard image $H(P_I)$. Later, perform inverse hadamard operation to the image $H(P_I)$ to obtain the original image (P_I) by the receiver. Hence, the decryption is performed.

The Proposed Hybrid Encryption Algorithm

Key Generation

The Sender (User X)

1. Select the private key $z_X \in [1,2,3,\ldots,p-1]$
2. Calculate the public key $Pu_X = z_X.G$
3. Calculate the initial key $Q_I = z_X.Pu_Y = (\alpha, \beta)$

4. Evaluate $Q_1 = \alpha \,.\, G = (q_{11}, q_{12})\ and\ Q_2 = \beta \,.\, G = (q_{21}, q_{22})$
5. Obtained 2×2 key matrix $Q_m = \begin{bmatrix} Q_1 \\ \\ Q_2 \end{bmatrix}$
6. Create a 4×4 self-invertible key matrix Q_n.

Key Generation

The Sender (User Y)

1. Select the private key $z_Y \in [1,2,3, \dots, p-1]$
2. Calculate the public key $Pu_Y = z_Y . G$
3. Calculate the initial key $Q_I = z_Y . Pu_X = (\alpha, \beta)$
4. Evaluate $Q_1 = \alpha \,.\, G = (q_{11}, q_{12})\ and\ Q_2 = \beta \,.\, G = (q_{21}, q_{22})$
5. Obtained 2×2 key matrix $Q_m = \begin{bmatrix} Q_1 \\ \\ Q_2 \end{bmatrix}$
6. Create a 4×4 self-invertible key matrix Q_n.

Encryption Process (User X)

1. Input: The Original image (P_I).
2. Apply Hadamard transform to the original image ($H(P_I)$).
3. Compute XOR operation with key Q_m and $H(P_I)$. The resultant image is R_I.
4. Divide the $R_I$image pixels into blocks of each size is four.
5. Set each block into (4×1) column vector.
6. Multiply each vector of the plain image ($P_{I1};\ P_{I2};\ P_{I3};\ \dots$) with a self-invertible key matrix $Q_n$and perform modulo 256 for each matrix value$C_{I1} = (\ Q_n \times\ P_{I1})\ mod\ 256$.
7. From the above ciphered vector($C_{I1};\ C_{I2};\ C_{I3};\ \dots$), the ciphered image C_I is constructed.

1) Decryption Process (User Y)

1. Divide the ciphered image (C_I) pixels into blocks of each size is four.
2. Set each block into (4×1) column vector.
3. Multiply each vector of a ciphered image ($C_{I1};\ C_{I2};\ C_{I3};\ \dots$), with a self-invertible key matrix Q_n and perform modulo 256 for each matrix value $P_{I1} = (\ Q_n \times\ C_{I1})mod\ 256$.
4. From the above deciphered vectors ($P_{I1};\ P_{I2};\ P_{I3};\ \dots$), the image R_I is generated.
5. Compute XOR operation with key Q_m and R_I. The resultant image is $H(P_I)$.

6. Apply inverse Hadamard transform to $H(P_I)$.
7. Finally, the original image (P_I) is retrieved.

Security Analysis

A successful cryptosystem should be able to withstand all kinds of known and regular attacks, including differential attack analysis, statistical analysis and therefore the key space should be large to fail the brute-force attack (Alvarez & Li, 2003). The current section provides the detailed view on security analysis and the test results to determine the efficiency of the algorithm proposed. Several regular tests are conducted on various images of the well-known database (Petitcolas, 2018), to determine the level of security and performance of the proposed cryptosystem. The images are used for this experiment are "Lena", "Peppers", "Barbara", and "Baboon" of 256 × 256 size.

Statistical Analysis

Statistical analysis (Shannon, 1949) will determine the ability to resist statistical attacks in terms of encryption efficiency. To verify whether the proposed encryption algorithm resist against statistical attacks, it includes various tests namely information entropy, correlation coefficient and histogram analysis.

1. Histogram Analysis

Histogram is a graph of an image which displays the intensity of each pixel and the distribution of the pixels intensity in that image. If the histogram is more uniform then it represents a good diffusion and hence the encryption is stronger which can resist the statistical attacks. In the Figure. 2 the original images "Lena", "Peppers", "Barbara", and "Baboon" and their corresponding original and cipher image histograms are displayed. It is noticed that the original and cipher images histograms are completely different and the cipher images histogram distributions are uniform. Clearly, from the encrypted image there is no useful information can be taken hence, ensuring high security. It indicates that the algorithm suggested has better capacity of opposing statistical attacks successfully.

2. Information Entropy

The image entropy is measured by the randomness of the pixels:

$$E(Z) = \sum_{j=0}^{m} P(z_i)\, log_2\, P(z_i) \qquad (29)$$

The variables for the aforementioned equation are as follows: m is the pixel grayscale quantity levels of an image, $z_i \in Z$ and $P(z_i)$ indicates the probability of outcome of pixel z_i. The image is achieved as true randomness if its maximum entropy reaches to 8. An efficient encryption scheme will make the cipher image entropy to maximum, the closer to the maximum is the higher to its efficiency. The Table 1 displays the entropy values of the images executed in the proposed encryption algorithm. From the Table 1, the cipher images entropies are very near to 8. Therefore, the entropy analysis attack is strongly resisted by the proposed algorithm.

Table 1 Information Entropy

Plain Image	Cipher Image
Lena	7.9986
Peppers	7.9990
Barbara	7.9981
Baboon	7.9992

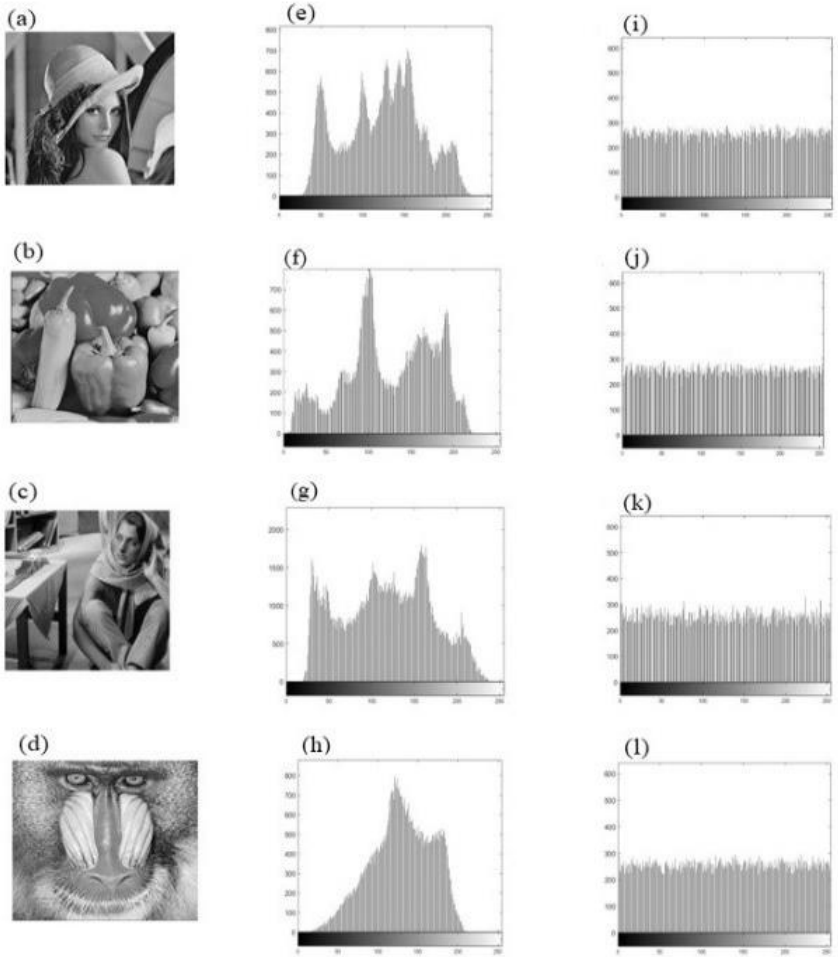

Figure 2 (a) - (d) Original Images; (e)-(h) Histograms of Original Images; (i)-(l) Histograms of Cipher Images

3. Correlation Analysis

The correlation between neighboring pixels in a plain image in all directions (horizontal, vertical and diagonal) is extremely high and typically close to 1. So, this correlation has to be decreased while encrypting and can be attained by using effective encryption method (Rostami et al., 2017). These methods should reduce the correlation of the pixels in the cipher image to 0. Here in this experiment, 2000 pairs of neighboring pixels are picked in all directions for calculation of correlation coefficient in both plain and cipher images. For calculating the correlation coefficient of both normal image and encrypted image, the below expression is used.

The correlation coefficient:

$$t_{uv} = \frac{cov(u,v)}{\sqrt{F(u)F(v)}} \tag{30}$$

$$cov(u,v) = E\{[u - E(u)] \quad [v - E(v)]\} \tag{31}$$

$$E(u) = \frac{1}{P}\sum_{i=1}^{P} u_i \tag{32}$$

$$F(u) = \frac{1}{P}\sum_{i=1}^{P}[u_i - E(u)]^2 \tag{33}$$

Where, u and v are two adjacent pixels, correlation coefficient is t_{uv}, Average values of u_i and v_i are $E(u)$ and $E(v)$, and number of pixels is P. Using this formula, the adjacent pixel's correlation is calculated and are noted in Figure 3 on all directions for both Lena's normal and encrypted images.

Table 2 Correlation coefficients of plain image and cipher image

Method	**Plain Image**				**Cipher Image**		
	Image	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Proposed Method	Lena	0.9345	0.9695	0.9195	-0.0015	-0.0027	0.0032
Hasheminejad & Rostami, 2019	Lena	0.9303	0.9628	0.9762	-0.0023	-0.0002	0.0045
Li et al., 2019	Lena	0.9689	0.9486	0.9228	-0.0029	-0.0031	-0.0037

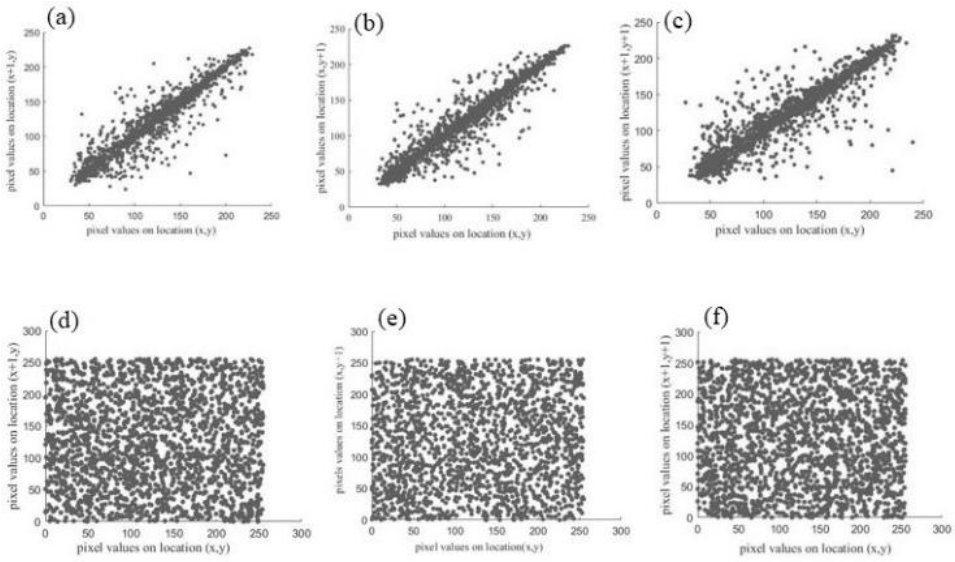

Figure 3 Adjacent pixel correlation of plain and cipher images of Lena. (a)-(c) Horizontal, vertical, diagonal directions in the plain image. (d)- (f) Horizontal, vertical, diagonal directions in the cipher image

The Table 2 displays the comparison of correlation with different methods for various images. It is evident from the results that the correlation between adjacent pixels in plain image is close to 1 and it is similar across other models. However, the correlation coefficient for adjacent pixels in encrypted image are different and for the current encryption algorithm it is close to 0. Hence, it can be inferred that the current method will significantly reduce the correlation coefficient between the pixels in an encrypted image and can be used for a secure communication.

4. Differential Attack

One of the main criteria to check the encryption technique security is differential attack analysis. A significant change should occur in the cipher image, if there is a minor change in the original image. The author in (Li et al., 2019) discussed, a change of 1-bit over the original image leads to a significant change in the cipher (resultant) image. Using two criteria's i.e. UACI and NPCR, such changes are assessed.

Number of Changing Pixel Rate (NPCR) =

$$N(w_1, w_2) = \sum_{i,j} \frac{F(i,j)}{L} \times 100\% \quad (34)$$

Unified Averaged Changed Intensity (UACI) =

$$U(w_1, w_2) = \sum_{i,j} \frac{|w_1(i,j) - w_2(i,j)|}{D \cdot L} \times 100\% \quad (35)$$

$$F(i,j) = \begin{cases} 0 \; if \; w_1(i,j) = w_2(i,j) \\ 1 \; if \; w_1(i,j) \neq w_2(i,j) \end{cases} \quad (36)$$

Where, D is largest pixel value, L is total pixels in cipher image, and $w1, w2$ are cipher images where only one pixel difference in their original image. The proposed algorithm is executed on the plain images and calculated the NPCR and UACI values as plotted in Table 3. The mean difference of two coupled cipher images are expressed by UACI, which are best at a lower value. However, NPCR centers around the exact number of pixels that changes the incentive in differential attack, which is better if the value is higher. According to (Wu et al., 2011) the UACI and NCPR expected values are 33.46% and 99.60% respectively. Table 4 shows the comparative analysis on different encryption techniques of NCPR and UACI values and states that the current proposed algorithm is better compared to remaining techniques. The algorithm introduced has accomplished the objectives required to avoid the differential attack.

Table 3 NPCR and UACI values

Plain Image	NPCR	UACI
Lena	99.6274	33.4232
Peppers	99.6643	33.0123
Barbara	99.5938	33.3554
Baboon	99.4945	32.8955

Table 4 Comparative analysis of NPCR and UACI on Lena image

	Proposed method	Hasheminejad& Rostami, 2019	Li et al., 2019	Rehman et al., 2016
NPCR	99.62	99.61	99.60	99.61
UACI	33.42	32.49	33.38	33.79

5. Key Space Analysis

The key space has to be sizeable enough to withstand any brute force attacks for a sensible encryption algorithm. Both public and private key of ECC are the keys of the proposed algorithm. The public key is accessible to everyone publicly, but the private one is secured and kept secretly. Also, the security of any algorithm depends on the key size. If the size of key is huge, it will be challenging to do a brute force attack. Theoretically, the security of ECC algorithm is completely dependent on the Elliptic Curve Discrete Logarithm Problem (ECDLP) difficulty. The ECDLP is the most challenging task in mathematics, and currently there is no method which can decipher it. The Diffie-Hellman key is Q_I which is found using the information of sender's private key z_X or the receiver's private key z_Y, The key

size for a EC prime parameter p which is of 256 bits(ECC-256) is 2256, which is sizeable enough to with stand the brute force attacks.

6. Key Sensitive Analysis

One important parameter for a good image encryption method is to have high sensitivity to all keys. There are two ways to measure it, one is, on changing the key value it should give a different encrypted image, other is, on changing the decryption key the plain image recovery should be impossible (Yu-Ling & Ming-Hui, 2013). In the current algorithm, using the correct key, encryption is performed on "Lena" image in the Figure. 4(a) and the encrypted image obtained is shown in Figure. 4(b). Now, with a small change in key is made and the new cipher image is obtained as shown in the Figure. 4(c) compared with the earlier cipher image Figure. 4(b) and the difference is calculated between two cipher images, the result is obtained in Figure. 4(d). The difference between the two cipher images are high and hence it proves that the current algorithm has high sensitivity, so that they can withstand against statistical and brute-force attack.

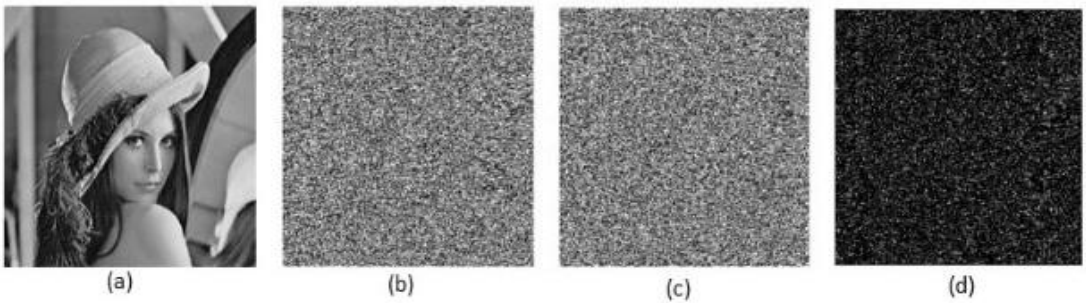

Figure 4 Key Sensitive Analysis. (a) Plain Image "Lena"; (b) Encrypted using Correct Key; (c) Encrypted using Modified Key; (d) The difference between (b) and (c)

7. Peak-Signal to Noise Ratio (PSNR)

The performance of the algorithm is checked by comparing the distortion between original image and decrypted image as well as original and encrypted image. The metric used to perform this encryption quality check is PSNR. The mathematical notation for PSNR is:

$$PSNR = 20\ log_{10}\frac{255}{\sqrt{MSE}} \tag{37}$$

Where, Mean Square Error (MSE) is obtained using the formula

$$MSE = \frac{1}{256\times256}\sum_{m=1}^{256}\sum_{n=1}^{256}[P(m,n) - E(m,n)]^2 \tag{38}$$

Where, $P(m,n)$ contains original image pixel values and $E(m,n)$ contains encrypted image pixel values. PSNR are MSE are inversely proportional to each other. Higher PSNR indicates that the original and decrypted images are almost identical at the same time the PSNR value calculated between original image and encrypted image should be less indicating high randomness between them. If the above said points for PSNR metric were met then the algorithm is efficient. Table 5 shows less PSNR values for the encrypted images so, the proposed algorithm is efficient.

Table 5 PSNR values

Plain Image	PSNR
Lena	8.4461
Peppers	8.7177
Barbara	8.8707
Baboon	9.4635

7. Computing Speed

Another quality metric for the encryption algorithm is its execution time. The execution time should be low for any efficient encryption algorithm. The proposed algorithm's execution time is calculated in MATLAB R2019a with this configuration 2.40GHz Intel(R) Core(TM) i7-5500U CPU and 8GB RAM. The comparison table is depicted in Table 6.

Table 6 Comparison on Execution Time

Method	Execution Time(sec)
Proposed Method	1.1133
Zhu S & Zhu C, 2018	1.4483
Chen et al., 2018	5.5567
Dawahdeh et al., 2018	1.2600

Conclusion

Data security is one of the most critical issues in the current situation. An efficient encryption technique is proposed in this paper. Here, a two level security is achieved using ECC, Hadamard and Hill cipher algorithm which makes it highly difficult to break the security. The key relies upon the ECC and it is difficult to resolve the ECDLP. Here, the Hadamard transformed image reduces the size of the image and helps in faster execution in calculating the encrypted image providing second level of security by completely hiding the image from the attackers. Also, Encryption and decryption using self-invertible key matrix, avoids redundant computation of finding inverse key during decryption that makes the process improve its speed. It improves the security and efficiency due to its self-invertible nature. The results of the experiment conducted and the comparison of

performances with various other methods show that the current cryptosystem provides high security, reliable, quick execution and higher efficiency compared to other EC related methods available in literature. This algorithm attains large key space, highly key sensitive and low correlation and can resist against differential and statistical attacks. As the proposed algorithm contains uncomplicated structure and quicker calculations, hence, suitable for all applications and is reasonable for small gadgets to embedded systems. Currently, the proposed algorithm is applied on the grayscale images in this paper. Due to the advantages of the proposed encryption algorithm in future, it may be tested, used on RGB images and other modes of information such as audio and video to improve the transmission efficiency.

References

Luo, Y., Zhou, R., Liu, J., Qiu, S., & Cao, Y. (2018). An efficient and self-adapting colour-image encryption algorithm based on chaos and interactions among multiple layers. *Multimedia Tools and Applications*, *77*(20), 26191-26217.

Song, J., & Lee, Y.H. (2021). Optical image encryption using different twiddle factors in the butterfly algorithm of fast Fourier transform. *Optics Communications*, *485*, 126707.

Kaur, G., Agarwal, R., & Patidar, V. (2020). Chaos based multiple order optical transform for 2D image encryption. *Engineering Science and Technology, an International Journal*, *23*(5), 998-1014.

Jiang, D., Liu, L., Zhu, L., Wang, X., Rong, X., & Chai, H. (2021). Adaptive embedding: A novel meaningful image encryption scheme based on parallel compressive sensing and slant transform. *Signal Processing*, *188*, 108220.

Wang, X., Feng, L., & Zhao, H. (2019). Fast image encryption algorithm based on parallel computing system. *Information Sciences*, 486, 340-358.

Simmons, G.J. (1979). Symmetric and asymmetric encryption. *ACM Computing Surveys (CSUR)*, 11(4), 305-330.

Huang, L., Cai, S., Xiong, X., & Xiao, M. (2019). On symmetric color image encryption system with permutation-diffusion simultaneous operation. *Optics and Lasers in Engineering*, 115, 7-20.

Luo, Y., Tang, S., Qin, X., Cao, L., Jiang, F., & Liu, J. (2018). A double-image encryption scheme based on amplitude-phase encoding and discrete complex random transformation. *IEEE access*, 6, 77740-77753.

Setyaningsih, E., Wardoyo, R., & Sari, A.K. (2020). Securing color image transmission using compression-encryption model with dynamic key generator and efficient symmetric key distribution. *Digital Communications and Networks*, *6*(4), 486-503.

Miller, V.S. (1986). Advances in Cryptology—CRYPTO'85 Proceedings. *Use of elliptic curves in cryptography*, 417-426.

Kobliz, N. (1987). Elliptic curve cryptography. *Mathematics of Computation*, *48*, 203-209.

Diffie, W., & Hellman, M. (1976). New directions in cryptography. *IEEE transactions on Information Theory*, 22(6), 644-654.

Shankar, K., & Eswaran, P. (2016). An efficient image encryption technique based on optimized key generation in ECC using genetic algorithm. In *Artificial intelligence and evolutionary computations in engineering systems*, 705-714.

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.

Khoirom, M.S., Laiphrakpam, D.S., & Themrichon, T. (2018). Cryptanalysis of multimedia encryption using elliptic curve cryptography. *Optik*, *168*, 370-375.

Tawalbeh, L.A., Mowafi, M., & Aljoby, W. (2013). Use of elliptic curve cryptography for multimedia encryption. *IET Information Security*, *7*(2), 67-74.

Hill, L.S. (1929). Cryptography in an algebraic alphabet. *The American Mathematical Monthly*, 36(6), 306-312.

Ismail, I.A., Amin, M., & Diab, H. (2006). How to repair the Hill cipher. *Journal of Zhejiang University-Science A*, *7*(12), 2022-2030.

Acharya, B., Rath, G.S., Patra, S.K., & Panigrahy, S.K. (2007). Novel methods of generating self-invertible matrix for hill cipher algorithm.

Khazaei, S., & Ahmadi, S. (2017). Ciphertext-only attack on d× d Hill in O (d13d). *Information Processing Letters*, *118*, 25-29.

Essaid, M., Akharraz, I., & Saaidi, A. (2019). Image encryption scheme based on a new secure variant of Hill cipher and 1D chaotic maps. *Journal of Information Security and Applications*, *47*, 173-187.

Harwit, M. (2012). *Hadamard transform optics*. Elsevier.

Qu, G., Meng, X., Yin, Y., Wu, H., Yang, X., Peng, X., & He, W. (2021). Optical color image encryption based on Hadamard single-pixel imaging and Arnold transformation. *Optics and Lasers in Engineering*, *137*, 106392.

Zheng, P., & Huang, J. (2018). Efficient encrypted images filtering and transform coding with walsh-hadamard transform and parallelization. *IEEE Transactions on Image Processing*, *27*(5), 2541-2556.

Prajwalasimha, S.N. (2019). Pseudo-Hadamard transformation-based image encryption scheme. In Integrated Intelligent Computing, *Communication and Security*, 575-583.

Devi, H.S., & Singh, K.M. (2020). Red-cyan anaglyph image watermarking using DWT, Hadamard transform and singular value decomposition for copyright protection. *Journal of Information Security and Applications*, *50*, 102424.

Wang, X., Liang, X., Zheng, J., & Zhou, H. (2019). Fast detection and segmentation of partial image blur based on discrete Walsh–Hadamard transform. *Signal Processing: Image Communication*, *70*, 47-56.

Anil K. Jain. (1989). *Fundamentals of Digital Image Processing*. Prentice Hall Inc., prentice hall international edition.

Li, L., Abd El-Latif, A.A., & Niu, X. (2012). Elliptic curve ElGamal based homomorphic image encryption scheme for sharing secret images. *Signal Processing*, 92(4), 1069-1078.

Alvarez, G., & Li, S. (2003). Cryptographic requirements for chaotic secure communications. *arXiv preprint nlin/0311039*.

Petitcolas, F.A. (2018). Public-domain test images for homeworks and projects. *URL: http://hompages. cae. wisc. edu/~ ece533/images*.

Shannon, C. E. (1949). Communication theory of security. *Bell System Technical Journal*, 28, 656-715.

Rostami, M. J., Shahba, A., Saryazdi, S., & Nezamabadi-pour, H. (2017). A novel parallel image encryption with chaotic windows based on logistic map. *Computers & Electrical Engineering*, *62*, 384-400.

Hasheminejad, A., & Rostami, M.J. (2019). A novel bit level multiphase algorithm for image encryption based on PWLCM chaotic map. *Optik*, *184*, 205-213.

Li, N., Sun, J., & Wang, Y. (2019). A novel memcapacitor model and its application for image encryption algorithm. *Journal of Electrical and Computer Engineering*, 2019.

Wu, Y., Noonan, J.P., & Agaian, S. (2011). NPCR and UACI randomness tests for image encryption. Cyber journals: multidisciplinary journals in science and technology, *Journal of Selected Areas in Telecommunications (JSAT), 1*(2), 31-38.

Rehman, A. U., Khan, J. S., Ahmad, J., & Hwang, S. O. (2016). A new image encryption scheme based on dynamic s-boxes and chaotic maps. *3D Research*, *7*(1), 7.

Yu-Ling, L., & Ming-Hui, D. (2013). A self-adapting image encryption algorithm based on spatiotemporal chaos and ergodic matrix. *Chinese Physics B*, *22*(8), 080503.

Zhu, S., & Zhu, C. (2018). Image encryption algorithm with an avalanche effect based on a six-dimensional discrete chaotic system. *Multimedia Tools and Applications*, *77*(21), 29119-29142.

Chen, J., Zhang, Y., Qi, L., Fu, C., & Xu, L. (2018). Exploiting chaos-based compressed sensing and cryptographic algorithm for image encryption and compression. *Optics & Laser Technology*, *99*, 238-248.

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.