ISSN: 0067-2904

Asymmetric Image Encryption Based on Singular Cubic Curve with Chaotic Map

Mohammed Jabbar Obaid*, Najlae Falah Hameed Al Saffar
Department of Mathematics, College of Computer Science & Mathematics, University of Kufa, Najaf, Iraq

Received: 10/1/2023 Accepted: 14/5/2023 Published: 30/5/2024

Abstract

Due to the potential security problem of the key management and distribution for the symmetric image encryption schemes, this paper proposes a new asymmetric image encryption method based on a singular cubic curve and chaos theory. The Diffie-Hellman algorithm is used to generate the initial values of a chaotic map by exchanging public keys. The image encryption process is carried out using a chaotic map, in which a random sequence of the same length as the image to be encrypted is generated using the standard map, and then bitwise XOR is used to create confusion in the image's pixels. Also, an Arnold cat map is used to change the location of pixels and diffusion in them is obtained. Gray images with size 256×256 are used in this algorithm. The simulation results and security analysis indicate that the scheme can withstand common attacks such as statistical attacks, differential attacks, and other attacks.

Keywords: Image Encryption, Asymmetric Image Encryption, Singular Cubic Curve, Chaotic Map.

تشفير الصور باستخدام المفتاح المعلن بالاعتماد على منحنى مكعب انفرادي النقط ودالة فوضوية

محمد جبار عبيد*، نجلاء فلاح حميد الصفار
قسم الرياضيات، كلية علوم الحاسوب والرياضيات، جامعة الكوفة، النجف، العراق

الخلاصة

نظرًا لمشكلة الأمان الخاصة بإدارة المفاتيح وتوزيعها لأنظمة تشفير الصور المتماثلة، تقترح هذه الورقة طريقة جديدة لتشفير الصور غير المتماثلة باستخدام منحنى مكعب مفرد النقط ونظرية الفوضى. تُستخدم خوارزمية ديفي هلمان لتوليد القيم الأولية لدالة فوضوية عن طريق تبادل المفاتيح المعلنة. يتم تنفيذ عملية تشفير الصور باستخدام دالة فوضوية، حيث يتم إنشاء تسلسل عشوائي بنفس حجم الصورة المراد تشفيرها باستخدام الدالة القياسية، ثم يتم استخدام عملية الجمع الثنائية من اجل تغيير قيم البكسلات في الصورة، وتستخدم دالة ارلوند لتغيير موقع البكسلات. تستخدم الصور الرمادية بحجم 256 × 256 في هذه الخوارزمية. تشير نتائج المحاكاة والتحليل الأمني إلى أن النظام يمكنه الصمود أمام الهجمات الشائعة مثل الهجمات الإحصائية والهجمات التفاضلية والهجمات الأخرى.

*Email: mohammedj.alghazali@student.uokufa.edu.iq

1. Introduction

The transmission of multimedia information over various networks has been greatly facilitated by the rapid expansion of network evolution. Most of these transmissions contents have occurred over an unsecured network which increases the likelihood of data loss, interception (i.e., illegal copying and distribution), and malicious tampering. As a result, there is a growing that concerns about the safety of multimedia data, particularly in the context of big data and cloud computing. As a result, there has been growing scholarly interest in the problem of image protection [1].

Cryptography is a mathematical method that is used to keep images and text safe from attackers and makes communication more secure. The encryption is performed by the sender, who converts the plain image into an encrypted image before sending it to the receiver via the Internet. Decryption is performed at the receiver's end, where the encrypted image is converted back to its plain image. There are two types of encryption methods that are used, namely symmetric encryption and asymmetric encryption. In symmetric encryption, both encrypting and decrypting use the same key. These algorithms work well and quickly, especially when a lot of information needs to be processed. However, symmetric encryption suffers from the difficulties of key management and distribution. The key must be sent securely over the network, but attackers can steal it while it is being sent. In fact, as the number of users grows, the number of keys will grow quickly which is hard on the network. Asymmetric (public-key encryption, PKE) encryption solves these problems by using two different keys public and private to encrypt and decrypt messages. It is difficult to obtain the private key from the public key. So, in this encryption, the private key does not need to be shared because the receiver already has one. So, the problem with how the key is shared does not matter for the encryption. It can also offer a digital signature that symmetric encryption cannot do. The two most complex mathematical problems in public-key encryption are the discrete logarithm problem and the factorization problem [2].

Miller [3] and Koblitz [4] developed elliptic curve cryptography (ECC) in 1985, a modern PKE that improves the efficiency of many techniques. Encryption experts have also found that ECC offers superior security with significantly smaller key sizes. Selecting the elliptic curve without a subexponentially technique has made the ECC more interesting because it allows us to solve the discrete logarithm problem. Compared to other algorithms, ECC's small parameters do not compromise its security. Reducing the need for memory, CPU time and network throughput make elliptic curve cryptography a more viable option. The Diffie-Hellman algorithm for elliptic curves [5] is widely used as a key exchange scheme in a variety of contexts. A singular cubic curve is a curve given by the Weierstrass equation $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$ that contains a singular point (0,0) and discriminant $\Delta = 0$ [6]. To propose new encryption systems, Koyama [7] replaced the elliptic curve with a singular cubic curve.

Numerous image encryption methods including DES, AES, RSA, etc. are based on number theory that was developed to address this security concern. Although these methods of encryption exist, they are not particularly effective when it comes to protecting visual content. This is because the images contain large amounts of data and have a high correlation between adjacent pixels and a high degree of redundancy, etc. [8].

In the past two decades, continuous and discrete chaotic dynamical systems were employed to construct cryptosystems [9] [10]. Chaos theory is used in cryptography for a variety of reasons that are inherent to the theory. These chaotic qualities include things like ergodicity, random-like behaviour, mixing properties, and sensitivity to the initial state and control settings. Chaos-based image encryption was implemented using a general confusion and diffusion architecture [11]. The pixels in an image are shuffled about in the confusion stage, but their values remain the same. The pixel values are changed in the diffusion stage to make it possible for a little change in one pixel to affect as many pixels as possible.

In this paper, the asymmetric image encryption algorithm is proposed based on singular cubic curves and chaotic maps such that singular cubic curve asymmetric cryptosystem and chaotic map are combined to get a new asymmetric cryptosystem. The proposed algorithm uses a singular cubic curve to generate the key which is then used as input for the chaotic maps through which the images are encrypted.

The rest of this work is organized as follows: In Section 2, the preliminary aspects of this paper are described. Section 3 gives details on the specifics of the proposed technique. Section 4 provides a comprehensive analysis of the proposed method. The conclusion of the research is provided in Section 5.

2. Preliminaries

This section gives context for and a precise explanation of the two research pillars: the singular cubic curve and the chaotic map.

2.1 Singular Cubic Curve (SCC)

A singular cubic curve over prime field F_p is defined by

$$(a,b): y^2 + axy \equiv x^3 + bx^2 \; mod \; p \qquad (1)$$

where $p > 3$ is a prime number and $a, b \in F_p$. A nonsingular part of a singular cubic curve is denoted by $S_p(a,b)$ which is defined as a set of all solutions $(x,y) \in F_p \times F_p$ in curve equation $E_p(a,b)$ that excludes the singular point $(0,0)$ and includes the point at infinity O_∞ [12].

Theorem 2.1 ([6]). Let $E_p(a,b)$ be a singular cubic curve. Then for each $t \in F_p$ there exists the map $\psi: F_p \to E_p(a,b)$ is defined by $\psi: t \to (x,y)$, $(x,y) = \left(t^2 + at - b, t^3 + at^2 - bt^2\right)$.

2.1.1 Addition Laws on Singular Cubic Curve

The addition law $\oplus$ on $S_p(a,b)$ is defined by the chord-and-tangent method in the case of elliptic curves nonetheless preserve in the case of singular cubic curves. For each point $P = (x_P, y_P) \in S_p(a,b)$, $P \oplus O_\infty = P$, and $-P = (x_P, -y_P - ax_P)$ the additive inverse of P, so $P \oplus (-P) = O_\infty$. For $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$. $P_1 \oplus P_2 = P_3 = (x_3, y_3)$ is calculated as follows:

$$P_3 = \begin{cases} (\lambda^2 + a\lambda - x_1 - x_2 - b, -\lambda(x_3 - x_1) - ax_3 - y_1), & \lambda = \dfrac{y_2 - y_1}{x_2 - x_1} \qquad if\ (x_1,y_1) \neq (x_2,y_2) \\ (\lambda^2 + a\lambda - x_1 - x_2 - b, -\lambda(x_3 - x_1) - ax_3 - y_1), & \lambda = \dfrac{3x_1^2 + 2bx_1 - ay_1}{2y_1 + ax_1} \qquad if\ (x_1,y_1) = (x_2,y_2) \\ O_\infty, & if\ x_1 = x_2\ and\ y_1 \neq y_2 \end{cases}$$

$S_p(a,b)$ is a finite Abelian group since this addition law exists. The scalar multiplication $k \otimes P$, $k \in F_p$ is defined as $k \otimes P = \underbrace{(P \oplus P \oplus P \oplus \ldots \oplus P)}_{k-times}$ [6] [12].

2.1.1 Singular Cubic Curve Discrete Logarithm Problem

Curves can be used to define the discrete logarithms problem for a finite group of the curve E that is generated by a point P. Finding an integer k such that $Q = k \otimes P$ is the discrete logarithm problem. The discrete logarithm of a point Q to P, where q is a different point on E, is the integer k [13]. In this section, the discrete logarithm problem is particularly proven on singular cubic curve $S_p(0,b): y^2 = x^3 + bx^2 mod\, p$.

Definition 2.2 ([6]). Order of elements in a group $S_p(0,b)$. The order of a point $P \in S_p(0,b)$ is the smallest positive integer k such that $k \otimes P = O_\infty$, it is denoted by $\#P$. The order of a $S_p(0,b)$ is the number of its points, it is denoted by $\#S_p(0,b)$.

Definition 2.3 ([14]). A finite group G is called cyclic if there exists an element $a \in G$ such that each element in G can be written as a^k, $k \in Z^+$. a will be a generator of G, denoted by $< a >$.

Definition 2.4 A finite group $S_p(0,b)$ is called a cyclic group if there exists a point $P \in S_p(0,b)$ generates each point in $S_p(0,b)$. The generator set will be denoted by $< \mathrm{P} >$ such that $< P >= \{P, 2 \otimes P, 3 \otimes P, \dots, k \otimes P\}, k \in Z^+$.

Remark 2.5 If P is a generator of $S_p(0,b)$, then $\#P = \#S_p(0,b)$.

Example 2.6 Consider the singular cubic curve $S_{11}(0,6): y^2 = x^3 + 6x^2 \ mod\ 11$ which has the corresponding group:

$$\{O_\infty, (3,2), (3,9), (5,0), (6,5), (6,6), (8,4), (8,7), (9,4), (9,7), (10,4), (10,7)\}.$$

Order of point $P = (6,5)$ is equal to 12, such that $2 \otimes P = (9,7)$, $3 \otimes P = (10,7)$, $4 \otimes P = (3,2)$,$5 \otimes P = (8,4)$, $6 \otimes P = (5,0)$, $7 \otimes P = (8,7)$, $8 \otimes P = (3,9)$, $9 \otimes P = (10,4)$, $10 \otimes P = (9,4)$, $11 \otimes P = (6,6)$, $12 \otimes P = O_\infty$. In the same way, the order of all the points can be calculated. Note that point P is a generator of group $S_{11}(0,6)$.

Proposition 2.7 A group $S_p(0,b)$ is a cyclic group of order $n, n \in Z^+$ if only if it has a point P of order n.

Proof. Forward part. Suppose $S_p(0,b)$ is a cyclic group of order n. This means $S_p(0,b)$ is generated by a point $P \in S_p(0,b)$, i.e., $S_p(0,b) =< P >= \{P, 2 \otimes P, 3 \otimes P, \dots, n \otimes P\}$. Since $\#S_p(0,b) = \#P$, therefore $\#P = n$. Then P is a point of order n.

Backward part. Suppose $\#S_p(0,b) = \#P$, $P \in S_p(0,b)$ for some $P \in S_p(0,b)$. The subset of $S_p(0,b)$ generated by P is given as follows:

$$< P >= \{P, 2 \otimes P, 3 \otimes P, \dots, n \otimes P\}$$

This contains n points. Thus $< P >= S_p(0,b)$. This result comes from Definition 2.4.

Definition 2.8 Let $S_p(0,b)$ be a finite cyclic group of order n. Let point P be a generator of $S_p(0,b)$, and $Q \in S_p(0,b)$. The Discrete Logarithm is the problem of finding an integer k such that $k \otimes P = Q$. The number k is called the Singular Cubic Curve Discrete Logarithm Problem of Q to point P.

Remarks 2.9:

1- A singular cubic curve $S_p(0,b): y^2 = x^3 + bx^2 mod\, p$ with a quadratic nonresidue b. For any integer β, $b \neq \beta^2 \ mod\, p$ will be used to get the best security in image encryption where the groups $S_p(0,b)$ and $\mathrm{F_p}$ are not isomorphic [15].

2- If $S_p(0,b)$ and $\mathrm{F_p}$ are isomorphic, then the difficulty level of solving the singular cubic curve discrete logarithm problem has the same level as solving the discrete logarithm problem of F_p. On the other hand, if $S_p(0,b)$ and F_p are not isomorphic, then the difficulty level of solving the singular cubic curve discrete logarithm problem has the same level as solving the elliptic curve discrete logarithm problem. That means, if a singular cubic curve is used as a tool, then a different approach to a public key cryptosystem will be involved.

2.1.2 Singular Cubic Curve Diffie–Hellman (SCCDH) Key Exchange

Two users on a network can exchange a shared session key in a secure manner using the SCCDH key exchange. The following describes how the key exchange is done:

- User A and user B agree on a singular cubic curve $S_p(0,b)$ and an element $g \in F_p$ such that it can be converted to a large-order point P using Theorem 2.1.
- User A chooses a secret integer n and calculates the point $A = n \otimes P \in S_p(0,b)$.
- User B chooses a secret integer m and calculates the point $B = m \otimes P \in S_p(0,b)$.
- Exchanged values of A and B between user A and user B.
- User B calculates $m \otimes A$ while user A calculates $n \otimes B$.
- They have now shared the $(k_1, k_2) = nm \otimes P$ value.

The SCCDH security is based on the difficulty of solving the singular cubic curve discrete logarithm problem that was defined in 2.8.

2.2 Chaotic Map

Chaotic maps are extremely sensitive to initial values and control factors. Any minimal alteration to the basic circumstances results in a remarkable deviation. This sensitivity severely restricts the ability to forecast. Initial conditions are used as a cryptographic key in chaos-based encryption techniques [16].

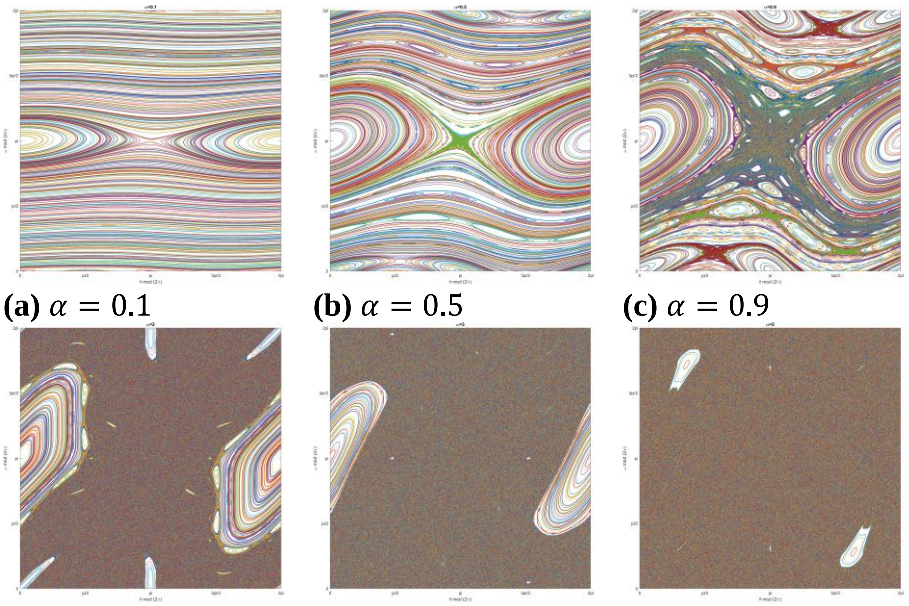
2.2.1 Standard Map

The standard map can be defined by the mathematical formula [17]:

$$\begin{cases} \omega_{n+1} = \omega_n + \alpha sin\theta_n \; mod\; 2\pi, \\ \theta_{n+1} = \theta_n + \omega_n \, mod\; 2\pi, \end{cases} \qquad (2)$$

where $(\omega_n, \theta_n) \in [0,2\pi) \times [0,2\pi)$, and the constant $\alpha > 0$. The level of chaos on the map increases proportionally with each successive increase in α value. Figure 1 shows different orbits on the standard map for different values of α. The dotted orbit is chaotic and develops in a large region of phase space as a random set of points, but the rest are periodic or quasi periodic.

Remark 2.10 The modulo 2π ($mod\; 2\pi$) is the remainder after dividing any real number by 2π. For example, 51.4 $mod\; 2\pi$ equals 1.1345 because 51.4/2π=8 with a remainder of 1.1345.



(a) $\alpha = 0.1$ **(b)** $\alpha = 0.5$ **(c)** $\alpha = 0.9$

(d) $\alpha = 2$ **(e)** $\alpha = 3$ **(f)** $\alpha = 5$

Figure 1: Orbits of the standard map of α values.

2.2.2 Arnold Cat Map

The Arnold cat map (ACM) is a discrete system with paths that move around in phase space and stretch and fold. Vladimir Arnold made the ACM in the 1960s. He used a picture of a cat , the size of the original grayscale image is $N \times N$ [18]. So the ACM will be as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = A \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & u \\ v & uv+1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \; mod \; N \; . \tag{3}$$

It is an area-preserving map where u and v are positive integers and $\det(A) = 1$. In the aftermath of a single execution of the Arnold cat map, the starting pixel location (x, y) will be located at (x', y').

Remark 2.11 The original image will resurface if the process is repeated enough times.

3. The Proposed Algorithm

In the proposed encryption algorithm, a SCC is used to give the sender and receiver a shared secret key with this key the control parameters of the chaotic map are set. Using the standard map, a random sequence of the same length as the plain image is made. After that applied the integer valued function for each value in this sequence, then bitwise XOR operation, denoted by $\bigoplus_{xor}$, is used to get confused in the pixels of the image. Finally, Arnold cat map is used to change pixels and gets diffusion in pixels of the image. So, the encrypted image $C = c_{i',j'}$ is obtained. The procedures of this algorithm are described below:

Key Creation Level: To create keys, the sender and receiver make the following steps:

step 1. Sender and receiver agree on a singular cubic curve $S_p(0, b)$ and an element $g \in F_p$ such that it can be converted to a large-order point P using Theorem 2.1, they applied SCCDH to get (k_1, k_2).

step 2. k_1 and k_2 will used to generate initial values ω_1, θ_1 and the constant α for the standard map, as $\omega_1 = k_1 mod\; 2\pi$, $\theta_1 = k_2 mod\; 2\pi$ and $\alpha = k_1 k_2 mod\; \pi$ +2. Then a vector L will be obtained by repeating the standard map; the number of elements in this vector is equal to the size of the image which is $N \times N$ as is shown below:

$L = (\omega_1, \omega_2, \omega_3, \dots, \omega_n, \theta_1, \theta_2, \theta_3, \dots, \theta_n), n = \frac{N \times \mathrm{N}}{2}$.

step 3. Round elements of the vector L to the closest integer k_1 and k_2 will be involved together in this step as follows:

$r_k = round(\omega_k \times 10^8) \times k_1 \; mod \; N,$ and $r_{k+n} = round(\theta_k \times 10^8) \times k_2 \; mod \; N,$ such that $round(\omega_k \times 10^8)$ and $round(\theta_k \times 10^8)$ is the process of rounding the numbers $\omega_k \times 10^8$ and $\theta_k \times 10^8$ to the closest integer, where $k = \left\{1,2,3, \dots, \frac{N \times N}{2}\right\}$.

step 4. Convert the vector $L = (r_1, r_2, r_3, \dots, r_{N \times N})$ to a matrix R of size $N \times N$. The process for converting a vector L is calculated by:

$$vm(L) = R = \begin{bmatrix} r_1 & r_{N+1} & r_{2N+1} & \cdots & r_{(N-1)N+1} \\ r_2 & r_{N+2} & r_{2N+2} & \cdots & r_{(\mathrm{N}-1)\mathrm{N}+2} \\ r_3 & r_{N+3} & r_{2N+3} & \cdots & r_{(\mathrm{N}-1)\mathrm{N}+3} \\ \vdots & \vdots & \vdots & & \vdots \\ r_N & r_{N+N} & r_{2N+N} & \cdots & r_{N^2} \end{bmatrix}_{N \times N} \tag{4}$$

step 5. Create the matrix A in terms of ACM; k_1 and k_2 are also involved in this construction, as: $A = \begin{bmatrix} 1 & k_1 \\ k_2 & k_1 k_2 + 1 \end{bmatrix}$.

Encryption Level: The sender encrypts the plain image M using a bitwise XOR operation between elements of matrix R and pixels of the image M where the matrix’s elements and the image's pixels are both converted to binary numbers. After that, the Arnold cat map is used to change the location of pixels as shown in the following steps:

step 1. $c_{i.j} = p_{i.j} \oplus_{xor} r_{i,j}$, where $p_{i,j}$ pixels of plain image M and $r_{i,j}$ elements of the matrix R.

step 2. Use ACM to change the location of $c_{i.j}$ as shown below:

$$\begin{bmatrix} i' \\ j' \end{bmatrix} = A \begin{bmatrix} i \\ j \end{bmatrix} = \begin{bmatrix} 1 & k_1 \\ k_2 & k_1k_2 + 1 \end{bmatrix} \begin{bmatrix} i \\ j \end{bmatrix} \; mod \; N, \qquad (5)$$

with $\det(A) = 1$, so the pixel location that was previously situated at (i, j) will be located (i', j'), As in Figure 1. Consequently, the cipher image C is acquired.

Decryption Level: The cipher image C will be decrypted by the receiver as shown in the following steps:

step 1. Use the Arnold cat map to return the location of pixels from (i', j') to (i, j) as shown below:

$$\begin{bmatrix} i \\ j \end{bmatrix} = A^{-1} \begin{bmatrix} i' \\ j' \end{bmatrix} = \begin{bmatrix} k_1k_2 + 1 & -k_1 \\ -k_2 & 1 \end{bmatrix} \begin{bmatrix} i' \\ j' \end{bmatrix} \; mod \; N \; , \qquad (6)$$

where A^{-1} is inverse of the matrix A.

step 2. Use $p_{i.j} = c_{i.j} \oplus_{xor} r_{i,j}$ then receiver gets plain image M.

3.1 Implementation

The implementation was performed using MATLAB Version 9.12.0.1884302 (R2022a) on a Lenovo Legion 5 laptop powered by an AMD Ryzen 5 4600H 3.00 GHz processor with 8GB of RAM. Four grayscale images with a size of 256×256 were used to implement the proposed algorithm. The plain images, cipher images, and decrypted images are all displayed in Figure 2.

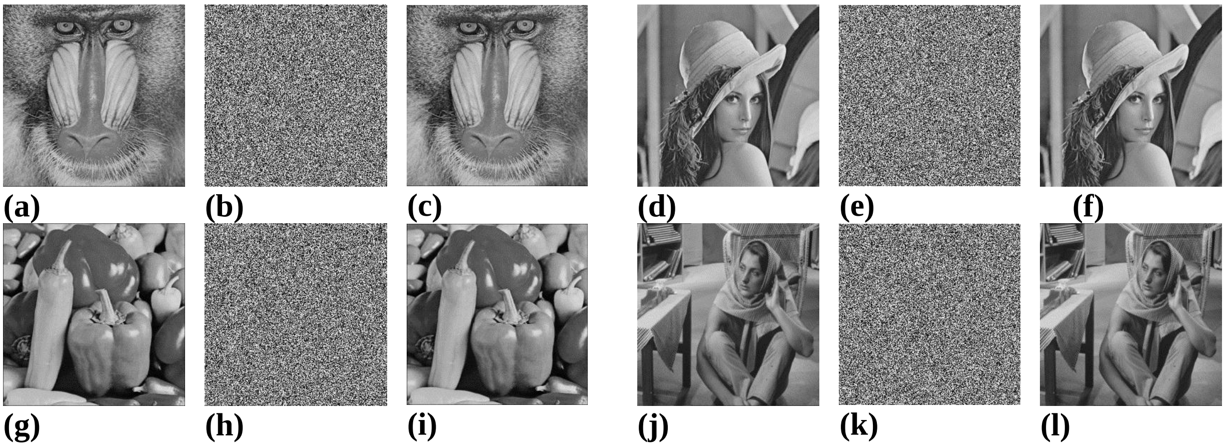


Figure 2: Implement the proposed algorithm: (a) Original image of Baboon, (b) Encrypted image of Baboon, (c) Decrypted image of Baboon, (d) Original image of Lena, (e) Encrypted image of Lena, (f) Decrypted image of Lena, (g) Original image of Peppers, (h) Encrypted image of Peppers, (i) Decrypted image of Peppers, (j) Original image of Barbara, (k) Encrypted image of Barbara, and (l) Decrypted image of Barbara.

4. Simulation results and analysis

A series of experiments are conducted to assess the effectiveness of the proposed image encryption scheme. This section presents the results of tests conducted to determine the efficacy of the proposed algorithm. Several tests are conducted on various images to determine the level of security and performance of the cryptosystem.

4.1 Key Space Analysis

A logical encryption scheme's key space must be large enough to withstand brute-force attacks [19]. The proposed algorithm employs the public and private keys of an SCC Cryptosystem. The private key is safeguarded and kept secret, whereas the public key is open to all. Furthermore, the key size determines the security of any algorithm. If the key is large, a brute force attack will be tough. The singular cubic curve discrete logarithm problem's (SCCDLP) difficulty. The SCCDLP is the most difficult mathematical problem. A 256-bit SCC prime parameter p (SCCC-256) has a key size of 256 which is large enough to survive brute force assaults.

4.2 Key Sensitivity Analysis

A secure image encryption technique should be sensitive to even minor changes in the decryption key. A good encryption algorithm must be extremely sensitive to the key. For testing sensitivity, the image of Lena is encrypted with the $k_1 = 489$ and $k_2 = 179$ keys and decrypted with the $k_1 + 1$ and k_2 keys as shown in Figure 3.

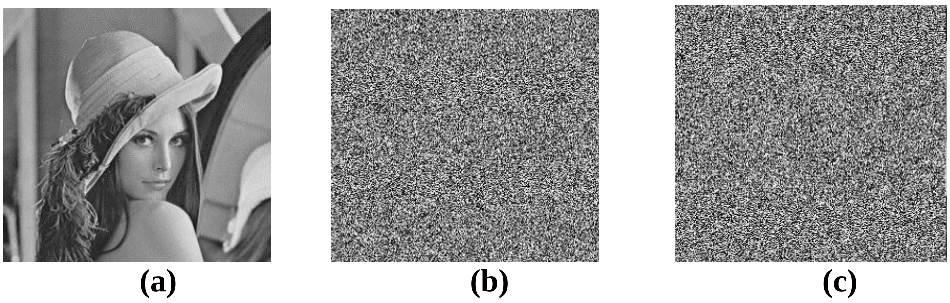


Figure 3: Key sensitivity test: (a) Original image of Lena, (b) Encrypted image of Lena using k_1 and k_2, and (c) Decrypted image of Lena using $k_1 + 1$ and k_2

4.3 Statistical Analysis

Statistical analysis can be used to determine whether an encryption system has the capacity to withstand an attack based on statistical data [20]. For statistical analysis, two methods of measurement are used: histogram analysis, and correlation coefficient analysis.

4.3.1 Histogram Analysis

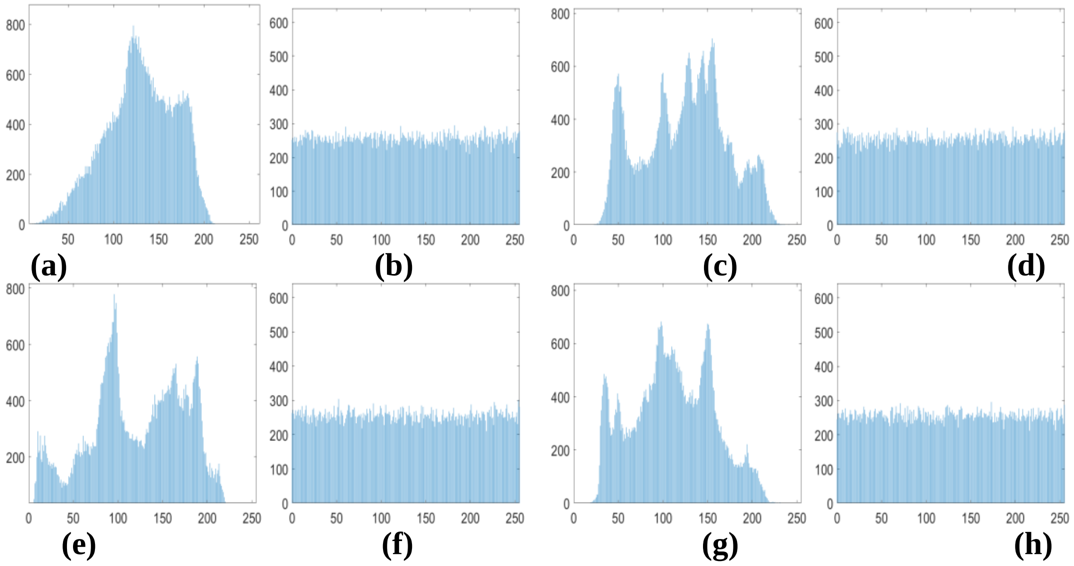
A histogram displays the variance in pixel brightness within an image. To ensure successful encryption, the image's histogram must be uniform and flat. As a result, there is nothing to disclose concerning the original photograph [21]. Figure 4 shows that the histogram of the encrypted image is uniform and clearly different from the histogram of the original image. Thus, it can protect itself from statistical assaults. In addition, the histogram's uniformity can be assessed using a chi-squared test [8] which is calculated as follows:

$$\chi^2 = \sum_{i=0}^{255} \frac{(z_i - z)^2}{z} , \tag{7}$$

where z_i represents the frequency with which a given pixel value appears in the image, and $z = \frac{N \times N}{256}$. Assuming a 0.05 significance level, $\chi^2_{255,0.05}$ equals 293.2478. If the value of χ^2 is less than 293.2478, then the histogram of the encrypted image is regarded as uniform. Table 1 demonstrates that the chi-square test for encrypted images yields small values.

Table 1: Chi-squared test.

Images	Baboon	Lena	Peppers	Barbara
χ^2	234.9609	243.5703	268.2109	228.4062



(a) **(b)** **(c)** **(d)**

(e) **(f)** **(g)** **(h)**

Figure 4: Histogram: (a) Original image of Baboon, (b) Encrypted image of Baboon, (c) Original image of Lena, (d) Encrypted image of Lena, (e) Original image of Peppers, (f) Encrypted image of Peppers, (g) Original image of Barbara, and (h) Encrypted image of Barbara.

4.3.2 Correlation Coefficient Analysis

One pixel in a plain image is typically and significantly associated with neighbouring pixels in the horizontal, vertical, and diagonal directions (usually close to 1). Therefore, a reliable image encryption technique might lessen this association [22]. In other words, it is anticipated that the correlation of the cipher image will be close to 0.

The formula for the correlation coefficient is as follows:

$$r_{xy} = \frac{\frac{1}{n}\sum_{i=1}^{M}(x_i - E(x))(y_i - E(y))}{\sqrt{\frac{1}{n}\sum_{i=1}^{M}(x_i - E(x))^2}\sqrt{\frac{1}{n}\sum_{i=1}^{M}(y_i - E(y))^2}}, \qquad (8)$$

where M is the total number of image pixels, x and y are two neighboring pixel values in a grayscale image, and $E(x)$ and $E(y)$ are their average values, respectively. Table 2 displays the correlation and spatial correlation values of the tested images. The results show that the proposed method successfully decouples the plain image from the encrypted image as well as the spatial correlation between them. Figure 5 depicts how the spatial correlation of the encrypted and plain images differs.

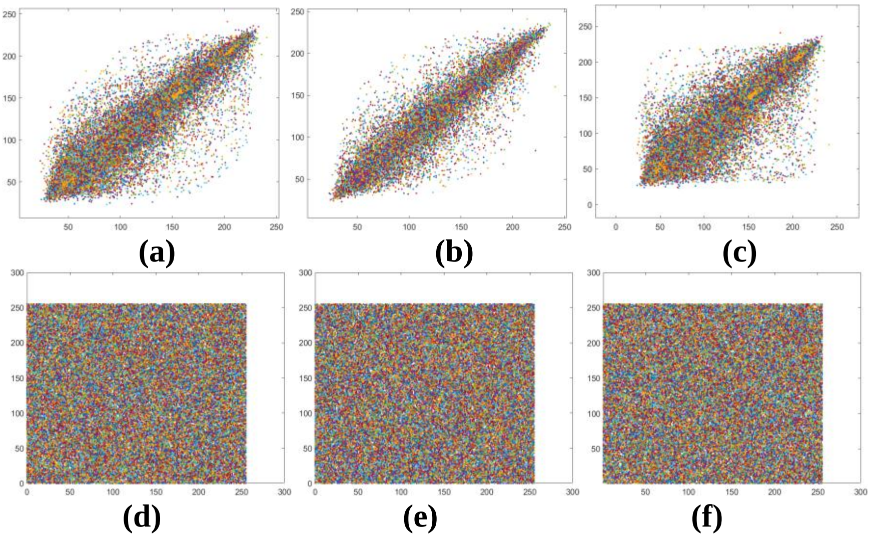


Figure 5: Correlation Coefficient: The horizontal, vertical, and diagonal correlation distributions of the original (a)–(c) and encrypted (d)–(f) Lena image, respectively.

Table 2: Correlation between plain and ciphered images

Images	**Horizontal**		**Vertical**		**Diagonal**	
	Plain	**cipher**	**plain**	**cipher**	**plain**	**cipher**
Lena	0.9456	0.0006	0.9727	-0.0035	0.9213	-0.0007
Baboon	0.8737	-0.0024	0.8261	0.0053	0.7843	0.0017
Peppers	0.9635	0.0026	0.9705	-0.0004	0.9365	-0.0027
Barbara	0.9450	0.0005	0.9547	0.0009	0.9043	0.0034

4.4 Information Entropy

The entropy of information is a useful metric for assessing the degree of disorder in a data set. The closer an encryption technique's information entropy is to number 8 (indicating security), the more difficult is for attackers to break the algorithm. Information entropy is defined as follows [23]:

$$H = -\sum_{i=0}^{2^M-1} p(\mathrm{s_i}) \log_2 p(s_i) \ , \qquad (9)$$

where M is the number of digits of the image pixel, and $p(s_i)$ is the probability of the sign s_i. Information entropy measurements are displayed in Table 3.

Table 3: Tests of information entropy

Images	Baboon	Lena	Peppers	Barbara
Entropy	7.9974	7.9973	7.9970	7.9975

4.5 Mean Square Error and Peak Signal to Noise Ratio

Peak signal to noise ratio (PSNR) and mean square error (MSE) are used to gauge the quality of image compression. The PSNR measures the cumulative squared error between the encrypted and original image, whereas the MSE measures the peak error. The MSE number determines how large the error is, where a large difference between the plain and encrypted images is indicated by a high MSE value, whereas a small difference between plain and decrypted images is indicated by a low MSE value. The accuracy of an image is instead determined by the PSNR. The difference between the plain and encrypted images grows larger as the PSNR decreases. The PSNR value is infinite between the plain and decrypted images. The MSE and PSNR equations are defined [24] as follows:

$$MSE_{PC} = \frac{1}{m \times n} \sum_{i=1}^{n} \sum_{j=1}^{m} \left(p_{i,j} - c_{i,j}\right)^2 \ , \quad MSE_{PD} = \frac{1}{m \times n} \sum_{i=1}^{n} \sum_{j=1}^{m} \left(p_{i,j} - d_{i,j}\right)^2 \qquad (10)$$

$$PSNR_{PC} = 10 \log_{10} \frac{n \times m}{MSE_{PC}}, \qquad PSNR_{PD} = 10 \log_{10} \frac{n \times m}{MSE_{PD}}, \qquad (11)$$

where P represents the plain image, C represents the encrypted image, and D represents the decrypted image, and m and n represent the height and width of the image, respectively. The MSE and PSNR data that are shown in Table 4 indicate that there is a considerable difference between the plain and encrypted images for the proposed approach but no difference between plain and decrypted images.

Table 4: MSE and PSNR for proposed algorithm

Images	MSE_{PC}	$PSNR_{PC}$	MSE_{PD}	$PSNR_{PD}$
Baboon	6960.4	9.7045	0	∞
Lena	7773.3	9.2248	0	∞
Peppers	8433.8	8.8706	0	∞
Barbara	7698.6	9.2667	0	∞

4.6 Differential Attack

Many attackers try to determine the encryption algorithm's weak point by altering a single plain image pixel and then comparing the resulting encrypted image to the original. An image encryption technique must react sensitively even if there is only a single bit difference to prevent scenarios in which attackers can identify any relevant associations between the cipher image and the encrypted modified images [25]. Tests called the Number of Pixels Change Rate (NPCR) and the Unified Average Changing Intensity (UACI) can be used to determine whether or not the algorithm is able to withstand this assault. The equations of UACI and NPCR are as follows:

$$UACI = \frac{1}{m \times n} \sum_{i,j} \frac{C_1(i,j) - C_2(i,j)}{255} \qquad (12)$$

$$NPCR = \sum_{i,j} \frac{D(i,j)}{m \times n} \ . \qquad (13)$$

Where C_1 and C_2 are two different cipher images encrypted using a different key and where $D(i,j)$ is defined as follows:

$$D(i,j) = \begin{cases} 1 & if\ C_1(i,j) \neq C_2(i,j) \\ 0 & if\ C_1(i,j) \neq C_2(i,j) \end{cases} . \qquad (14)$$

The UACI and NPCR theoretical values are 33.46% [26] and 99.6094% [27], respectively. Table 5 shows that the proposed image encryption algorithm can withstand differential attacks because the NPCR and UACI values are close to their theoretical values.

Table 5: shows that the suggested technique is secure against a differential attack.

Images	Baboon	Lena	Peppers	Barbara
UACI	33.4191	33.4286	33.5052	33.5874
NPCR	99.6017	99.6323	99.5865	99.6353

4.7 Results and Comparisons

Table 6 and Table 7 indicate that the proposed algorithm is good at resisting a variety of attacks as shown by comparison with other encryption techniques.

Table 6: Comparison between the proposed algorithm and the other algorithms in terms of spatial correlation

Methods	**Lena**			**Peppers**			**Baboon**		
	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Proposed	0.0006	-0.0035	-0.0007	0.0026	-0.0004	-0.0027	-0.0024	0.0053	0.0017
Ref. [28]	-0.0003	-0.0037	0.0020	-0.0002	0.0020	0.0048	-	-	-
Ref. [29]	-	-	-	0.0045	-0.0042	-0.0018	0.0026	-0.0043	0.0034
Ref. [30]	-0.0003	0.0020	-0.0017	0.0025	-0.0050	0.0018	-0.0015	0.0034	0.0015
Ref. [1]	-0.0023	-0.0017	-0.0022	0.0044	-0.0008	0.0019	-0.0042	0.0010	-0.0010
Ref. [24]	-0.0012	-0.0014	0.0016	-0.0013	-0.0016	-0.0013	0.0010	0.0010	-0.0013

Table 7: Comparison between the proposed algorithm and the other algorithms in terms of MSE_{PC}, $PSNR_{PC}$, UACI, NPCR and Entropy

Methods	$\boldsymbol{MSE_{PC}}$		$\boldsymbol{PSNR_{PC}}$		**UACI**		**NPCR**		**Entropy**	
	Baboon	Lena	Baboon	Lena	Baboon	Lena	Baboon	Lena	Baboon	Lena
Proposed	6960.4	7773.3	9.7045	9.2248	33.4191	33.4286	99.6017	99.6323	7.9974	7.9973
Ref. [28]	-	-	-	-	-	33.27	-	99.59	-	7.9971
Ref. [24]	6901.7	7835.4	9.7412	9.1902	33.42	33.65	99.61	99.61	7.9976	7.9974
Ref. [31]	7254.20	7.9974	9.5248	9.2392	33.6430	33.6124	99.6438	99.6641	7.9993	7.9993
Ref. [32]	-	-	-	-	33.4702	33.4994	99.6368	99.6216	7.9967	7.9972

5. Conclusion

In order to protect the information contained in digital images, this work introduces an asymmetric image encryption algorithm based on the SCC with chaotic mappings. The proposed algorithm allows for the safe handling and transfer of keys. The proposed algorithm’s unpredictability is increased by altering both the pixel values and the positions of each pixel in the image. Based on actual results and algorithm evaluations, the security analysis demonstrates that the proposed method can resist numerous forms of cryptanalysis including brute force assaults, statistical attacks, and differential attacks.

References

[1] H. Liang, G. Zhang, W. Hou, P. Huang, B. Liu and S. Li, “A novel asymmetric Hyperchaotic image encryption scheme based on elliptic curve cryptography,” *Applied Sciences,* vol. 11, no. 12, p. 5691, 2021.

[2] D. Vamsi and P. R. CH, “Hybrid image encryption using elliptic curve cryptography, Hadamard transform and Hill cipher,” *Webology,* vol. 18, no. 1, pp. 2357-2378, 2021.

[3] V. S. Miller, “Use of elliptic curves in cryptography,” in *Conference on the theory and application of cryptographic techniques*, Springer, 1985, pp. 417-426.

[4] N. Koblitz, “Elliptic curve cryptosystems,” *Mathematics of Computation,* vol. 48, no. 177, p. 203–209, 1987.

[5] R. Merkle and M. Hellman, “Hiding information and signatures in trapdoor knapsacks,” *IEEE Transactions on Information Theory,* vol. 24, no. 5, pp. 525-530, 1978.

[6] J. H. Silverman, The arithmetic of elliptic curves, vol. 106, New York, NY: Springer New York, 2009.

[7] K. Koyama, “Fast RSA-type schemes based on singular cubic curves y 2 + axy ≡ x 3 (mod N),” in *International Conference on the Theory and Applications of Cryptographic Techniques,* Springer, 1995, pp. 329-340.

[8] K. A. Patro, B. Acharya and V. Nath, “Secure, lossless, and noise-resistive image encryption using chaos, hyper-chaos, and DNA sequence operation,” *IETE Technical Review,* vol. 37, no. 3, pp. 223-245, 2020.

[9] A. A. Abdallah and A. K. Farhan, “A new image encryption algorithm based on multi chaotic system,” *Iraqi Journal of Science,* vol. 63, no. 1, pp. 324-337, 2022.

[10] S. Du and G. Ye, “IWT and RSA based asymmetric image encryption algorithm,” *Alexandria Engineering Journal,* vol. 66, pp. 979-991, 2023.

[11] Y. Alghamdi, A. Munir and J. Ahmad, “A lightweight image encryption algorithm based on chaotic map and random substitution,” *Entropy,* vol. 24, no. 10, p. 1344, 2022.

[12] D. Husemöller, Elliptic curves, vol. 111, New York, NY: Springer, 2004.

[13] A. J. Menezes and N. Koblitz, Elliptic curve public key cryptosystems, vol. 234, New York, NY: Springer, 1993.

[14] J. B. Carrell, Groups, matrices, and vector spaces: A group theoretic approach to linear algebra, New York, NY: Springer, 2017.

[15] L. C. Washington, Elliptic curves: Number theory and cryptography, Boca Raton: Chapman & Hall/CRC, 2008.

[16] R. Matthews, “On the derivation of a “chaotic” encryption algorithm,” *Cryptologia,* vol. 13, no. 1, pp. 29-42, 1989.

[17] A. Aragoneses, A. Kapulkin and . A. K. Pattanayak, “Permutation entropy of indexed ensembles: Quantifying thermalization dynamics,” *arXiv preprint arXiv:2211.00503,* 2022.

[18] G. Peterson, “Arnold’s cat map,” *Math Linear Algebra,* vol. 45, pp. 1-7, 1997.

[19] Y. Luo, L. Cao, S. Qiu, H. Lin, J. Harkin and J. Liu, “A chaotic map-control-based and the plain image-related cryptosystem,” *Nonlinear Dynamics,* vol. 83, no. 4, pp. 2293-2310, 2016.

[20] I. A. Taqi and S. M. Hameed, “A new color image encryption based on multi chaotic maps,” vol. 59, no. 4B, pp. 2117-2127, 2018.

[21] S. M. Hameed, H. A. Sa’adoon and M. Al-Ani, “Image Encryption Using DNA Encoding and RC4 Algorithm,” *Iraqi Journal of Science,* vol. 59, no. 1, pp. 434-446, 2018.

[22] M. J. Rostami, A. Shahba, S. Saryazdi and H. Nezamabadi-pour, “A novel parallel image encryption with chaotic windows based on logistic map,” *Computers &amp; Electrical Engineering*, vol. 62, pp. 384-400, 2017.

[23] Z. Man, J. Li, X. Di and O. Bai, “An image segmentation encryption algorithm based on hybrid chaotic system,” *IEEE Access,* vol. 7, pp. 103047-103058, 2019.

[24] M. Ramzan, T. Shah, M. M. Hazzazi, A. Aljaedi and A. R. Alharbi, “Construction of S-boxes using different maps over elliptic curves for image encryption,” *IEEE Access,* vol. 9, pp. 157106-157123, 2021.

[25] Y. Wu, J. . P. Noonan, S. Agaian and S. Member, “NPCR and UACI randomness tests for image encryption,” *Cyber journals: multidisciplinary journals in science and technology, Journal of Selected Areas in Telecommunications (JSAT),* vol. 1, pp. 31-38, 2011.

[26] Z. K. Obaid and N. F. Al Saffar, “Image encryption based on elliptic curve cryptosystem,” *International Journal of Electrical and Computer Engineering (IJECE),* vol. 11, no. 2, p. 1293, 2021.

[27] M. Khan and F. Masood, “A novel chaotic image encryption technique based on multiple discrete dynamical maps,” *Multimedia Tools and Applications,* vol. 78, no. 18, pp. 26203-26222, 2019.

[28] M. Maazouz, A. Toubal, B. Bengherbia, O. Houhou and N. Batel, “FPGA implementation of A Chaos-based image encryption algorithm,” *Journal of King Saud University - Computer and Information Sciences,* vol. 34, no. 10, pp. 9926-9941, 2022.

[29] G. Liu, W. Li, X. Fan, Z. Li, Y. Wang and H. Ma, “An image encryption algorithm based on discrete-time alternating quantum walk and Advanced Encryption Standard,” *Entropy,* vol. 24, no. 5, p. 608, 2022.

[30] B. Jasra, M. Saqib and A. H. Moon, “Image Encryption Using Logistic-Cosine-Sine,” *Journal of Theoretical and Applied Information Technology* , vol. 99, no. 16, 2021.

[31] I. Yasser, F. Khalifa, M. A. Mohamed and A. S. Samrah, “A new image encryption scheme based on hybrid chaotic maps,” *Complexity,* vol. 2020, pp. 1-23, 2020.

[32] L. Liu, Y. Lei and D. Wang, “A fast chaotic image encryption scheme with simultaneous permutation-diffusion operation,” *IEEE Access,* vol. 8, pp. 27361-27374, 2020.